هل تريدون فعلا أن تصبحو مبرمجين محترفين و لكن لاتدرون من أين تبدؤون ؟
بتعبير آخر :هل مللتم من الدروس المعقدة التي لا تفهمون منها شيئا ؟)

إنه يوم حظم
لقد وجدتم اليوم دوروسا في البرمجة خاصة بالمبتدئين خاص, كل الخصاص بالمبتدئين

في الحقيقة لا يوجد عيب في كوننا مبتدئين فالكل مرة بهذه المرحلة للوصول إلى الإحتراف ,أنا أيضا .

و الواجب منا و الذي أعتبره سهلا و هو أنه يجب تعليمكم كل شيئ و ذلك بدءا من :

إلى أية برامج نحن بحاجة إليها لنبدأ البرمجة ؟-
ما هي اللغة التي سنبدأ بها (يبدو أن ذلك ظاهر من العنوان)-
لكن لماذا لا نختار لغات إخرى للبدأ؟و أولا ماذا نعني بلغة؟
كيف نصنع برامج مثل الألعاب ,و النوافذ...إلخ-

في الحقيقة دروسي و التي تعتبر مجهودا شخصيا سنقسمها إلى ثلاثة أقسام أو لنسميها أشطرا و هي كالتالي:
شطرين للتعلم و هما ينقسمان لشرط الأساسيات و شطر+ التقنيات المتقدمة
و الشطر الثالث سنخصصه للتطبيق و ذلك بصنع أول لعبة عن+ طريق برامج و بتطبيق ما تعلمناه من الشطرين السابقين

و من هنا تنتهي مقدمتنا و هناك بعض الأشياء التي سأذكر بها و هي الصبر فالصبر مفتاح الفرج و هذه الخصلة هي من أكبر و أهم خصال المبرمج المحترف فلذلك إذا كنتم عازمين على التعلم فأنصحكم بالتريث و الصبر فكل شيئ يأتي متدرجا

و الآن فلنبدأ درسنا الأول و هو عبارة عن مقدمة كبيرة حول البرمجة تحت عنوان :

قلتم "برمجة "؟
فماذا نعني ببرمجة ؟
نبرمج في أي لغة ؟
و هل البرمجة صعبة ؟

للمبتدئين C السلام عليكم !!و مرحبا بكم في درسي للبرمجة ب

سأكون مرشدكم أو بالأحرى لنقل أستاذكم إذا أردتم طول فترة الدرس .
لقد قمت بعدة مواضيع مفيدة للمنتدى لكنني في الحقيقة لم أضع أي موضوع للتعلم فقط للتحميل أو ما شابه و يعتبر هذا الموضوع الذي أمام أعينكم أول درس أقوم به دون نقل من أناpaz78؟ إسمي أو إسم المستخدم الذي هو
لقد تكلمت عن نفسي كثيرا,فلـ نتكلم عنكم.
أنتم داخل هذا الموضوع من أجل هدف محدد و واضح :تعلم البرمجة .أنتم لا تعرفون شيئا في البرمجة ,أو أنتم لا تعرفون ما هي حتى ,و لكن البرمجة بلغة سي ماذا يعني هذا ؟هل هذه اللغة هي المناسبة للبدء بالبرمجة ؟و هل لديكم الخبرة الكافية للبدء بالبرمجة؟ و هل يمكننا أن نفعل بها كل شيئ هذا الموضوع له هدف محدد هو الإجابة عن هذه الأسئلة السخيفة و لكن أيضا المهمة جدا.

البرمجة ماهي ؟

سنبدأبالسؤال السهل و القاعدي من كل الأسئلة ,لكن إذا كنتم تعرفون كل هذا فأنصحكم بقراءة هذه السطور ,فلن تموتو لو قرءتموهم ,سنبدأ من الصفر في هذا الدرس لذلك سأجيب عن السؤال
ماذا يعني "برمج"؟
و بكل بساطة "برمج "تعني عمل برنامج إلكتروني .و البرنامج يطلب من الكومبيوتر فعل أعمال و حاسوبكم مملوء بعدة برامج مثلا :
-الآلة الحاسبة برنامج
-معلج النصوص هو برنامج
-ألعاب الفيديو هي عبارة عن برامج ...

بإختصار شديد البرامج توجد في كل مكان و هي تسمح بفعل أي شيئ على الحاسوب



اللعبة المشهورة half life 2 المبرمجة بلغة C++

**إنتباه أنا لم أقل أنا هذا سهل بل فقط قلت أن كل هذا ممكن و لكن كونوا على يقين أن كل هذا يتطلب عملا كبيرا و كثيرا.
بما أنكم مبتدئون فلن نبدأ بعمل ألعاب 3دي ,سيكون إنتحارا بمعنى الكلمة
... سنبدأ بأشياء بسيطة ومن بينها كيف نظهر كتابة في الشاشة درجة بدرجة سنتعلم أمورا بسيطة لكن مهمة و سنبدأ بعمل برامج متدرجة الصعوبة .
لكن هل تعرفون قصة لغة c/c++**

**الحاسوب هو آلة غريبة, هذا أقل ما يمكن قوله .فلا يمكننا أن نتحدث إليه سوى عن طريق 1 و0.
لو أردت كتابة مثلا 3+5 بلغة الكمبيوتر فستعطينا شيئا كهذا :
0010110110010011010011110
أنا أخترع هاهاهاه... فأنا لا أعرف الترجمة الإلكترونية عن ظهر قلب)
فالحاسوب لا يعرف إلا هذه اللغة ..إذا فمشكلنا الأول هو :كيف يمكننا أن نتكلم مع الكمبيوتر بطريقة أسهل من 1 و 0؟ الذي ترونه فوق هو لغة الكمبيوتر أو ما يسمى ب langage binaire
حاسوبكم لا يتكلم الفرنسية و لا العربية ,لكن من لهم خبرة في هذا المجال إخترعوا لغات أخرى اللواتي سيترجمن في ما بعد إلى اللغة التي يفهمها الكومبيوتر ,لكن الأصعب و هو هو عمل برنامج لترجمة هذه اللغات .لحسن الحظ هذا البرنامج هذا موجود**

و لسنا بحاجة لكتابته (أوف..).
في الحقيقة سنستعين بمثل هذه البرامج لكتابة مثلا "إفعل 3+ 5".
و هذا الأمر سيترجم بفعلها إلى مثلا:
"0010110110010011010011110".
و لنقم برسم بياني يبين ما قلناه :

برنامجكم بعد كتابته بطريقة بسيطة على الشكل التالي مثلا:"إفعل الأمر 3+5".



قيام برنامج ترجمة اللغات بتحويل برنامجكم المكتوب بلغة برمجة إلى لغة بفهمها الكمبيوتر.



نحصل على برنامج يفهمه الكمبيوتر بلغته الخاصة :" 101110101001"

في الخانة الأولى نكتب بلغة برمجة برنامجا و هذه اللغة الغريبة تسمى "لغة من المستوى العالي "
و نسمي لغة من مستوى عالي لغة قريبة من من اللغة الحقيقية مثل العربية و يوجد العديد من اللغات ذات المستوى العالي مثل
C
C++
Java
Visual Basic
Delphi
إلخ إلخ....
من الآن فصاعدا سوف أعطيكم بعض الكلمات الخاصة بمجال البرمجة و لنبدأ إنطلاقا من رسمنا البياني :
code source: و هو كود البرنامج الذي نريد إنشاءهو الذي سيتم تحويله إلى لغة الحاسوب

compilateur : وهو البرنامج المتخصص بتحويل الكود سورس إلى اللغة التي يفهمها الحاسوب
ملحوظة :يوجد لكل لغة برنامج تحويل خاص بها,وهذا جد منطقي فاللغات مختلفة فلا نحول سي++بنفس الطريقة التي نحول بها ديلفي...

l'exécutable: و هو البرنامج المحول إلى لغة الحاسوب و الذي يتم تشغيلة كبرنامج الرسام,الآلة الحاسبة ....

و الآن فلنعد لرسمنا البياني و نكتبه بالمصطلحات الجديدة :

برنامجكم بعد كتابته بلغة من المستوى العالي على الشكل التالي مثلا:"إفعل الأمر 3+5".



compilateur



l'exécutable(program.exe)

لماذا نختار لغة سي للبرمجة ؟

أظن أن الجواب عن هذا السؤال يكمن في كون لغة سي ستعلمكم أكثر حول البرمجة و حول كيفية إشتغال جهازكم . كما أن سي لغة أكثر شعبية من اللغات الأخرى و هي الأكثر إستخداما في برمجة قسم كبير من البرامج .
و كذلك للبرمجة بلغة سي لن تضطرون لشراء برنامج و لا إستعمال كراك و سنرى ذلك في الدرس الثاني .

سأعتبر لغة سي أول لغة برمجة بالنسبة لكم لكن إذا كنتم تعرفون قليلا أو كثيرا عن البرمجة فلن يضركم أن تعيدو من الصفر.

هل البرمجة صعبة ؟

البرمجة ليست بالشئ الصعب إذا تحلينا بالصفات التالية :
+الصبر و الإنتظار: فالبرنامج لا يشتغل مع الضغطة الأولى, يجب أن نتعلم الصبر!!
+المنطق :يجب عليك التفكير و التفكير ففي بعض الأوقات سنقوم ببعض التمارين.
+الهدوء:فلا يجب أخذ المطرقة و نبدأ بضرب الكومبيوتر, فليس بهذه الطريقة سيشتغل البرناج .

أسئلة الدرس :

لغة برمجة من المستوى العالي ,هي
1- لغة برمجة صعبة معقدة تشبه لغة الحاسوب
2- لغة سهلة تشبه كثيرا اللغة الحقيقية //المستعملة //
3-لغة لا يستعملها إلا المحترفون و الذين لديهم دراية بهذا المجال

البرنامج المسؤول عن تحويل الكود سورس إلى لغة الحاسوب يسمى:

1-Le binarisateur

2-Le brumisateur

3-Le compilateur

و ها قد إنتهى درسنا لليوم و نرجو من المشرفين الغاليين تثبيت الموضوع و ذلك حتى يستفيد الكل من الموضوع , و أعلمكم أن هذه الدروس ستكون متجددة أسبوعيا و ذلك لضيق و قت الفراغ للكتابة

و سيكون هذا الموضوع هو الرئيسي بالنسبة للمواضيع الأخرى أي هنا سيتم و ضع روابط الدروس الأخرى

و شكراااااا على قراءتكم الموضوع

إنتظروا الدرس القادم تحت عنوان :

ماذا أحتاج للبرمجة ؟

مع تحياتي

C / الدرس الثاني: ماذا أحتاج للبرمجة؟...../دورة الدرس الثاني

C الدرس الثالث : برنامجكم الأول بلغة سي

الدرس الرابع : عالم من المتغيرات

C. الدرس الخامس : العمليات الحسابية في لغة

الدرس السادس: الإحتمالات

تمارين تطبيقية...

الدروس مترجمة من الأخ الفاضل : مصدر الدروس m@teo 21 من
موقع www.siteduzero.com

أشهد أن لا إلـه إلا **الـلـه** و أن **محمدا** رسول **الـلـه**

بعد الدرس الأول الذي تضمن بلا بلا بلا بلا (و لكن مهمة و مفيدة)و ها قد بدأنا ندخل في صلب الموضوع و في هذا الدرس سنجيب على السؤال التالي:

ماهي البرامج التي نحن بحاجة إليها لكي نبدأ بالبرمجة ؟

لن يكون في هذا الدرس ما هو صعب و لكن سنأخذ كل الوقت لنتعرف على كيفية عمل مختلف برامج البرمجة .
فاستغلوا الفرصة لأننا في الدرس القادم سنبدأ حقا بالبرمجة و لن يكون لنا الوقت لأخذ القيلولة .
عنوان الدرس :

الأدوات التي يحتاجها المبرمج
...DEv C++ يمكنكم إختيار
...Visual C++ أو
...Code::Blocks أو أيضا
Xcode... و بالنسبة للماكنتوش

إذا بالنسبة لكم ما الأدوات التي يحتاجها المبرمج ؟
لو كنتم تابعتم الدرس السابق, فستعرفون واحدا على الأقل !!

هل عرفتم الجواب ؟

...
...
...
لم تعرفوه ؟

هذا البرنامج الغريب , compilateur نعم, إن الأمر يتعلق بال
الذي يمكننا من ترجمة الكود سورس إلى اللغة التي يفهمها
الحاسوب !
compilateur و كما قلت لكم في الدرس السابق أنه يوجد عدة
وسنرى بأن إختيار "المترجم " لن يكون صعبا بالنسبة لنا .C للغة
نحن المبتدئون .
إذا, إلى ماذا يحتاج أيضا ؟
لن أدعكم تخمنون كثيرا, إليكم أقل المعدات التي يجب أن تتوفر
عند المبرمج لكي يبدأ :
+Un éditeur de texte معالج للنصوص : و ذلك لكتابة الكود
سورس للبرنامج, بصفة عامة برنامج مثل Bloc-Notes تحت
الويندوز أو "vi" و هو الأحسن و .يفيان بالغرض تحت اللينكس
التوفر على معالج نصوص ذكي, يلون الكود بنفسه, و الذي
سيساعدكم في مراجعة الكود و تصحيح الأخطاء بكل سهولة .
+Un compilateur : و ذلك لترجمة الكود سورس للغة
الحاسوب .

+Un débugger و هو يمثل متعقب الأخطاء و (تذكروه جيدا)
سيساعدكم في تصحيحها (للأسف لم يتم إختراع أي
"مصحح"حتى الآن).

من الآن لدينا طريقتين للإختيار و العمل :

-إما أن نقوم بإستعمال البرامج السابقة متفرقة .و هي الطريقة
الصعبة و المعقدة و لكنها تعمل تحت اللينكس فقط و بشكل
جيد, و هناك عدة مبرمجين يفضلون هذه الطريقة .لن أشرح
أكثر هنا . لكنني سأريكم الطريقة الثانية و هي أبسط و أسهل و
في متناول الجميع .

-و إما أن نستعمل البرامج مجتمعة في برنامج واحد "3 en 1"
"3in1" و الذي يجمع éditeur de texte, compilateur و
débugger

هذه البرامج التي تجمع الثلاثة في واحد نسميها ب IDE.

في الحقيقة يوجد أكثر من بيئة عمل و تطوير , و قد تجدون
بعض الإحتيار في الإختيار لكن شيئ واحد مؤكد و صحيح و هو
أنه يمكنكم فعل أي شيئ تريدونه في أي برنامج (بالنسبة لنا).

الأفضل من بينهم كلهم ؟ ما هو IDE

**فبه بدأت و إذا لم أختر code::blocks بالنسبة لي فإنني أفضل فليس لعدم قوته لكنه خاص visual C++ الآخرين مثل بالمحترفين في هذا المجال فأنا شخصيا أعتبره الأقوى .
visual C++لكن و بما أنكم مبتدئون فلن أجبركم على إختيار فلن تكونوا قادرين على على إستغلال قوته الكاملة .**

**و Dev C++ لذلك فلم يبقى سوى خيارين و هما
قد لاقى IDE هو Dev C++ و لمعلوماتكم فإن . Code::Blocks
شعبية كبرى لمدة و لكن لم يتلقى أي تحديث منذ مدة كبيرة .
للبدء , و لكنه ليس Code::Blocks لذلك فإنني أنصحكم ب
الذي تختارونه ستكونون قادرين على IDE أمرا.فكيف ما كان
فعل الكثير من الأشياء .**

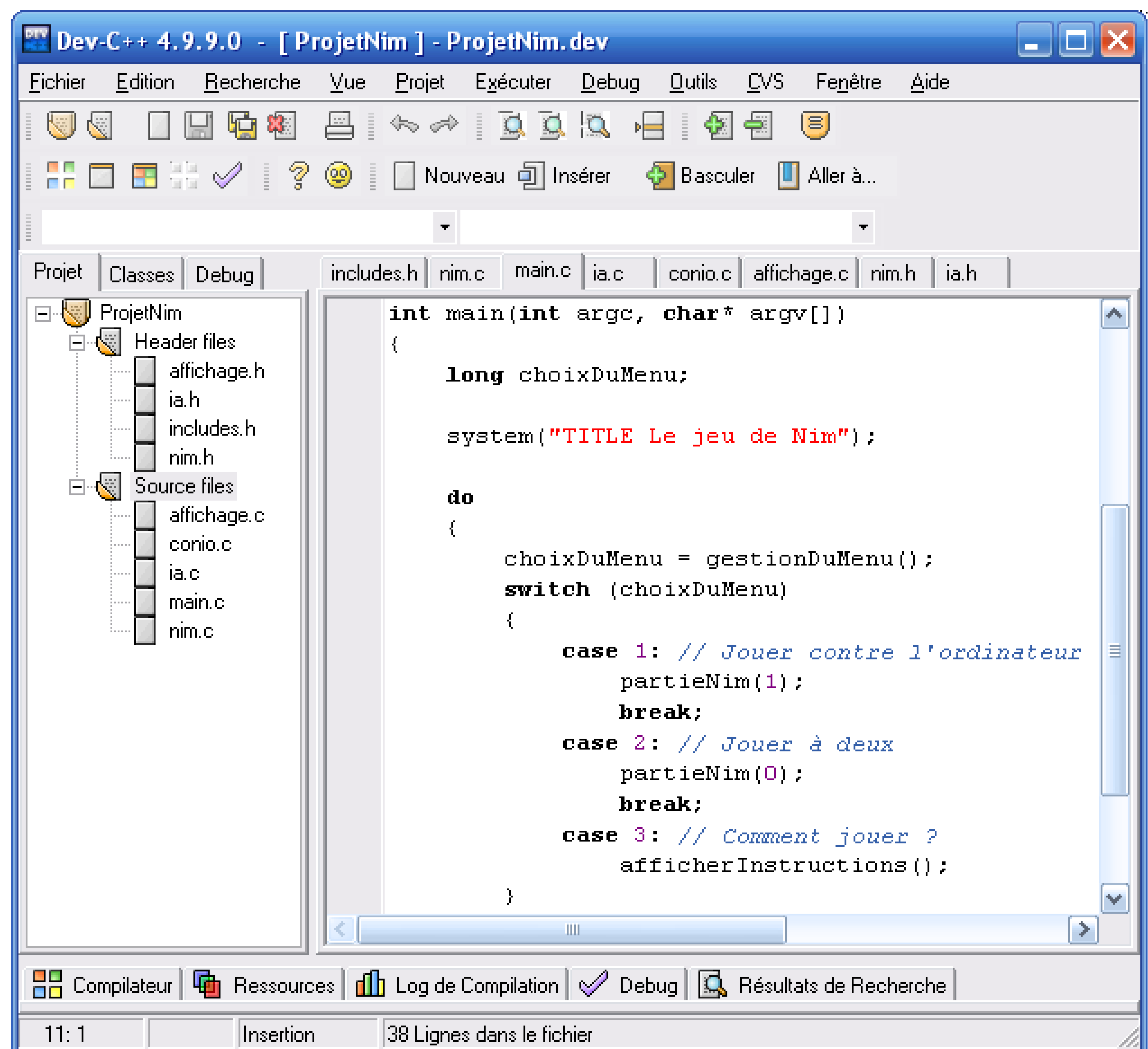
**في بقية هذا الدرس سأقوم بعرض البرامج 3 .
لكن لاحظو أن هذه البرامج لا تستخدم إلا تحت الويندوز , إلا
و الذي يشتغل إيضا تحت اللينكسcode::blocks .**

و ماذا أفعل إذا كنت أستخدم الماكنتوش ؟


**و "Xcode" يسمى ب IDE إذاكنتم تحت الماك فهناك
و سنرى كيفية Mac OS الذي يوجد في سي دي التنصيب ل
إشتغاله في نهاية الدرس .**

يمكنكم إختيار DEv C++..

**مجاني . و هو الأكثر (IDE) هو بيئة عمل و تطوير DEv C++
شهرة .و لكنه ليس محدثا , لذلك فاليوم ينصح بإستعمال
Code::Blocks (مجاني أيضا, موضح في الأسفل).**



و ككل البرامج ,توجد عدة نسخ. و الصور التي سأعرضها هي من النسخة 4.9.9.0 .

مثل هذه البرامج تتطور بسرعو فإذا كنتم من مستخدمي نسخة متقدمة فلا تخافو... فإن عمل الرنامج و دوره لن يتغير

و لتحميله لا يسعنا سوى الدخول لموقع البرنامج وتحميله بكل سهولة دون الإحتياج لكراك أو سيريال ...

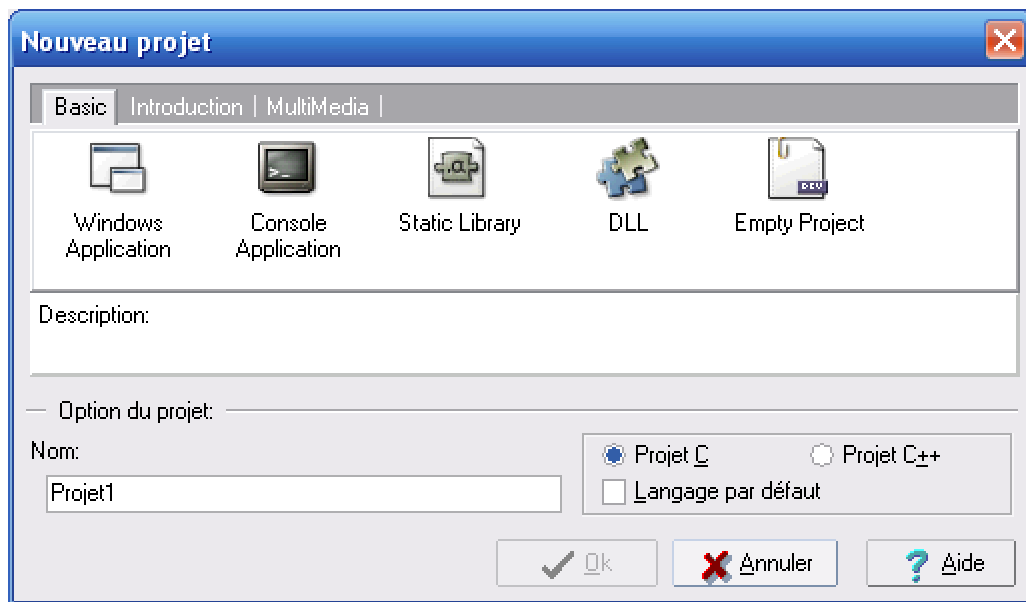
Site web for Dev C++

كيفية العمل عليه ؟

شغل البرنامج .

في المرة الأولى سوف يسألكم حول اللغة و حول إنشاء ملفات خاصة, أنصحكم بتركها كما هي .
!حا نحن في البرنامج لكن لايوجد شيئ؟؟
للبرنامج و لفعل ذلك new projet يجب طلب مشروع جديد
: إذهب إلى القائمة الرئيسية و إتبع الأوامر
"Fichier / Nouveau / Projet"
أو
File / New / Project""

: بعد ذلك ستخرج لكم نافذة مثل



+ إضغط على "Console Application" .فلن نبدأ بعمل برامج بالنوافذ بل سنقتصر فقط على برامج الconsole وهي تظهر على نوافذ تشبه كثيرا الدوس .
+ثم إختاروا "Projet C"
+ثم علموا "Langage par défaut"
+ و من ثم أعطوا إسما لمشروعكم
+ثم ok

بعد ذلك سيطلب منكم مكان الحفظ (أنصحكم بعمل مكان خاص لمشاريعكم و برامجكم)

و كما نلاحظ فإن البرنامج يكتب بعض الأشياء لن أقوم بشرحها إلى في الدرس القادم و سيكون بالتفصيل الممل .

```c
#include <stdio.h>
#include <stdlib.h>

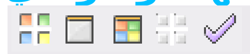
int main(int argc, char *argv[])
{

  system("PAUSE");
  return 0;
}
```

و للذين لم يعرفوا كيفية إنشاء مشروع جديد فقد أنشأت فيديو لكن باللغة الفرنسية
من هنا

أهم القوائم في البرنامج ...

لن أتكلم عن كل القوائم فهي لنفسها شارحة لكنني سأتكلم عن ما أراه مهما و هو في هذه الصورة :



سأبدأ من اليسار إلى اليمين

1- الأيقونة الأولى compilateur : وهي تقوم بدور الترجمة .

2-الأيقونة الثانية : و هي لرؤية البرنامج (الشيئ الذي حصلنا عليه بعد كتابة الكود سورس).

3-الأيقونة الثالثة : و هي تعمل عملا الإثنتين السابقتين في

نفس الوقت .

ملاحظة : بالنسبة للفيجوال سي++ لن أقوم بشرحه لأننا لن نستعمله

لذلك سأمر مباشرة لل code::blocks

يمكنكم إختيار code::blocks

code::blocks و هو IDE حر و مجاني .
لكنه يعتبر جديدا و ليس له نفس شهرة الديف سي ++ , و لكنني و مع ذلك أعترف بأنني كنت مندهشا من هذا البرنامج الرائع و أنصحكم بإستعمال (نصيحة أب لإبنه).
الوحيد الذي يشتغل تحت IDE بالإضافة إلى ذلك فإنه ال الويندوز و الينكس في نفس الوقت ,لكن و للأسف لأصحاب الماكنتوش لا توجد نسخة خاصة به .

بالنسبة لمستخدمي الفيستا فسأقوم بشرح إستخدام هذا البرنامج في فرصة أخرى ...

نأتي للتحميل

بالنسبة للتحميل فهو من هنا بالنسبة لأصحاب الويندوز

كيفية التنصيب سهلة فقط دع الكل كما هو و شغل البرنامج

بالنسبة لشرح القوائم فهو كالتالي:

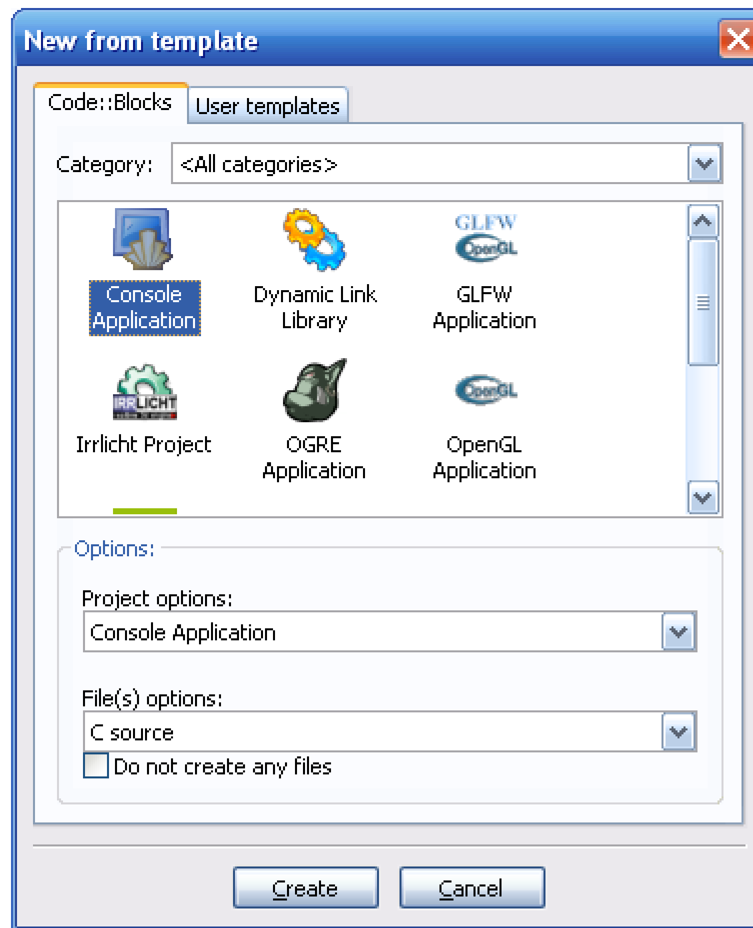


+) بالنسبة لهذه القوائم فهي مثل شرح قوائم البرنامج السابق (Dev C+

كيفية إنشاء مشروع جديد في الكود بلوكس إتبع الأمر

File / New Project.

و حددوا "Console application" ستفتح نافذة إختاروا فيها
"File Options : C Source"



. لإنشاء المشروع "Create" و من ثم إضغطوا على
و هنا أسضا سيطلب منك حفظ الملفات) و أعيد نصحكم
بإستخدام مجلد خاص بمشاريعكم و برامجكم).

إنتهى درسنا لليوم ...
بالنسبة لأصحاب الماكنتوش فلم أستطع توفير الشرح بنفسي
لذلك سأطلب من الأصدقاء و الأعضاء عمله (بالنسبة لي
فسأبحث في الأنترنيت عن شرح مناسب وشكراIIII).

أسئلة الدرس :

1/ ما هو العنصر الذي لا ينتمي إلى ال IDE:

-Le compilateur
-Le débugger
-Le freezer

2/ماذا نعني ب console:
-برنامج يشتغل على البلاي ستايشن
-برنامج يعمل في بيئة تشبه الدوس
-برنامج يعمل في نافذة مع عدة أزرار و خصائص

. إنتهى الدرس و أنتظر إستفساراتكم و آراءكم حول الموضوع
: و إنتظروا الدرس القادم بعنوان

برنامجكم الأول بلغة C

مع تحياتي

الدرس الأول : هل تريدون تعلم البرمجة و لا تدرون من أين تبدؤون ؟
الدرس الأول/C إذن إدخلو ../دورة لغة

أشهد أن لا إلـه إلا **الـلـه** و أن **محمدا** رسول **الـلـه**

لقد هيأنا الأرضية حتى الآن , وها قد حان الوقت لكي نبدأ بالجد, فما رأيكم ؟

فهذا هو الهدف الحقيقي من الدرس !! في نهايته , ستكونون قادرين على كتابة جملة أو كلمة على الشاشة .
إذا , فبرنامجكم سيكون من أسود و أبيض و لن يصلح إلا لكي يقول لكم "السلام عليكم", يمكنكم القول أن هذا ليس بشيئ الكبير و لا المهم لكنكم في النهاية ستكونون جد فرحين لوصولكم لهذه الدرجة فالكل مرة منها للوصول إلى الإحتراف .

و لنبدأ متى تشاؤون (أقصد أنتم :إذا أردتم تعلم البرامج فها قد بدأنا نغوص في صلب الاالموضوع).

عنوان الدرس :

كونصول أو نافذة (console ou fenêtre).
أقل ما يمكن كتابته(minimum de code).
كتابة شيئ على الشاشة .
التعليقات (les commentaires).

كونصول أو نافذة (console ou fenêtre).

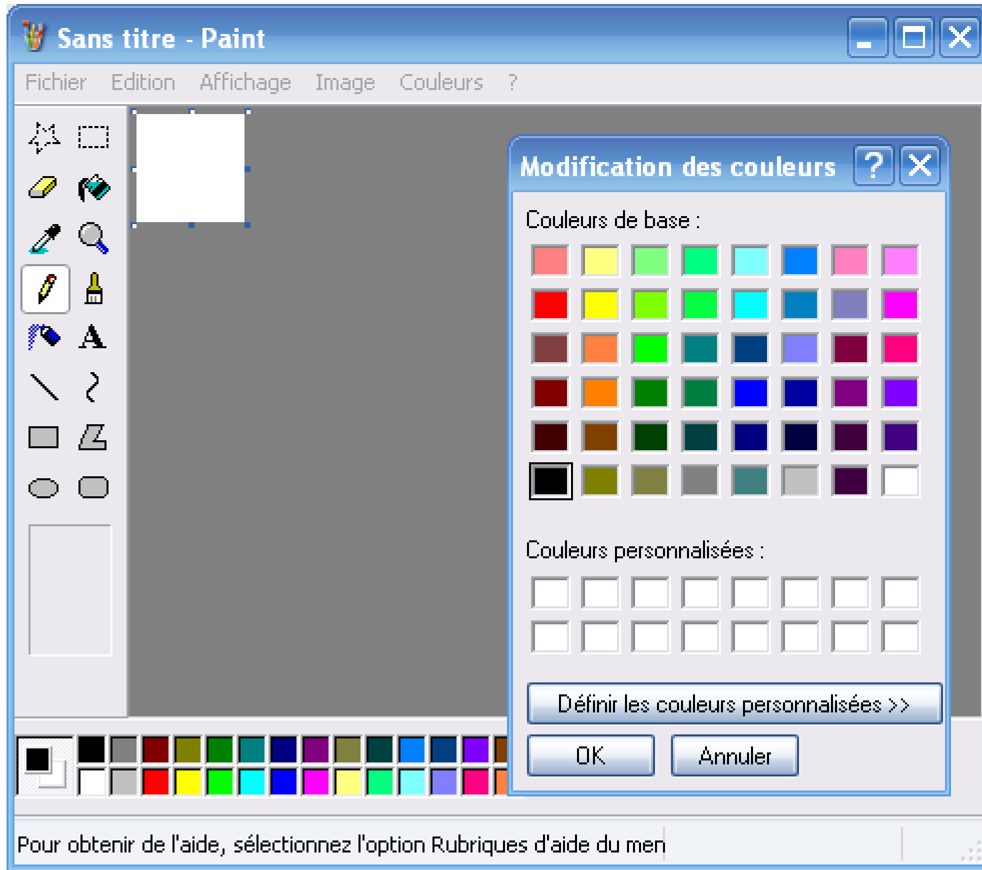
كونصول أو نافذة (console ou fenêtre)؟
لقد تكلمنا سابقا و بالتحديد في الدرس الثاني .. على أن سيطلب منا نوع البرنامج الذي (Visual أو Dev) IDE برنامجنا نريد عمله , و قد قلت لكم أن تقوموا بإختيار كونصول .

يجب علينا أن نعلم أن هناك نوعان من البرامج , لا أكثر:

+ البرامج بالنوافذ : البرامج التي تستخدم النوافذ للظهور.
+برامج الكونصول : البرامج التي تستعمل نافذة تشبه الدوس.

البرامج بالنوافذ

و هي البرامج التي نعرفها .
و هذا مثال حي لهذه البرامج و التي تعرفونها حقا :



هذه إذن هي البرامج بالنوافذ .
أعتقد أنكم تريدون كلكم إنشاء مثل هذه البرامج ...ممم؟
ستتمكنون من ذلك في ما بعد .

بطبيعة الحال , إنشاء برامج بالنوافذ بلغة سي شيئ ممكن !!!. لكن... بما أننا مبتدؤون, سيكون هذا شيئا جد صعب. كنصيحة مني للبدء يجب البدء بعمل برامج الكونصول.

برامج الكونصول

لمعلوماتك: برامج الكونصول هي الأولى التي ظهرت . في هذا الزمن, الحاسوب لم يكن يتعمل إلا بالأبيض و الأسود فلم يكن في هذه الأوقات قادرا على فتح نوافذ كما نفعل اليوم .

و مرت الأوقات . و ظهر النظام الجديد الذي سيقلب العالم و هو windows فكما يدل إسمه على النوافذ فقد كان هذا النظام النظام أشهر الأنظمة و ما يزال في الصدارة و ذلك لسهولة التعامل معه و لأنه لا يستعمل إلا النوافذ ...و في هذه الأثناء أصبحت الكونصول مهمشة تقريبا !
هي أنتم هناك , لا تستديرو فأنا أتحدث معكم ...

لدي خبر سعيد الكونصول لم تمت فقد أعاد نظام اللينكس لهذا

التراث إذا لم نقل "أصل كل شيئ " و هذه صورة للكونصول
على نظام لينكس :

```
2.2.5_appli.html        3.2.7.css           3.6.8.html
2.2.5.css               3.2.8.css           3.6.9.html
2.2.6_appli.html        3.2.9_appli.html    ancres.html
2.2.6.css               3.2.9.css           base.php
2.3.10_appli.html       3.3.10.html         cible_formulaire.php
2.3.10.css              3.3.11.css          cible.html
2.3.11_appli.html       3.3.12.css          design1.css
2.3.11.css              3.3.13_appli.html   erreur_paragraphe.html
2.3.12.css              3.3.13.css          essai2.css
2.3.13.html             3.3.14_appli.html   essai.css
2.3.14.css              3.3.14.css          images
2.3.15.html             3.3.15.css          tests_design.html
2.3.16.css              3.3.1.css           traitement.php
2.3.17.css              3.3.2.html
2.3.18.css              3.3.3.css
[root@ns1 exemples]# cd ..
[root@ns1 xhtml-css]# ls
anims                 css.php               images       pseudoformats.php
annexes               design.php            images.php   qcm.php
autres                exemples              index.php    tableaux.php
boites_partiel.php    formatage_partiel.php intro.php    texte.php
boites_partie2.php    formatage_partie2.php liens.php    xhtml.php
conclusion.php        formulaires.php       listes.php
[root@ns1 xhtml-css]#
```

. إذا الآن عندكم فكرة إلى ماذا تشبه الكونصول
لكن الكونصول تطورت و أصبحت قادرة على إظهار الألوان
. كالكتابة بالأحمر و الأزرق و عمل خلفيات دون الخلفية السوداء
(لكن هذا فقط على اللينكس).

لكن إذا كنت على الويدوز ؟ألا توجد كونصول ؟

بلا توجد و لكن إنها ...."مخفية " إذا صح القول

يمكنكم الوصول إليها عن طريق
"Démarrer / Accessoires / Invite de commandes"

أو

"Démarrer / Exécuter" ثم نركب الأمر "cmd".

و ها هي الكونصول الفريدة من نوعها للويندوز :

```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows XP [version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\Mateo>_
```

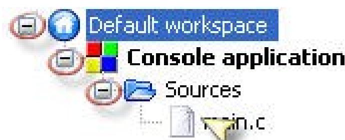
إذا هنا سنقوم بعمل برامجنا لكن كونو على يقين أنه عندما يكون لدينا المستوى لعمل برامج النوافذ سنرى كيف يتم ذلك ؟؟.

أقل ما يمكن كتابته(minimum de code).

لإنشاء أي برنامج يجب علينا كتابة شيئ و هو ما يسمى ب minimum de code.
في الحقيقة هذا الكود لن يقوم بأشياء مهمة نحن المبتدؤون لكنه يعتبر أساسيا.

فأنا لحد الآن code::blocks أنا الآن سأتحدث على أنكم إخترتم مازلت أشتغل به .

بعد فتح مشروع جديد كما شرحت من قبل إتبعو ما يلي لنرى هذا minimum de code كيف هو ..!!!



نضغط هنا مرتين

main.c

```
#include <stdio.h>
#include <stdlib.h>

int main()
{
    printf("Hello world!\n");
    return 0;
}
```

هذا هو ما يسمى ب
minimum de code

التحليل

فالنبدأ بالسطرين الأولين :
كود PHP:

```
Code : C
#include <stdio.h>
#include <stdlib.h>
```

إنها سطور متخصصة و التي نراها في أول البرنامج . تسمى هذه السطور ب directives de préprocesseur (إسم معقد أليس كذلك).

تدل على الفعل أضف في include فكما تعلمون أن الكلمة العربية .. إذن فهذه البرامج تطلب من البرنامج إضافة ملفات إلى المشروع .
فكما نلاحظ هناك سطران أي ملفين مضافين .أسماء هذه الملفات هي stdio.h و stdlib.h
بإختصار شديد هذه الملفات تحتوي على أكواد سورس مهيئة من قبل يعني نحن لن نقم بإعادة كتابتها و سنرى فيما بعد ما

يسمى ب **librairies** .(لن أثقل عليكم فلنتابع الدرس).

دون هذين السطرين كتابة شيئ على الشاشة أو إظهاره شيئ مستحيل .

لنمر الآن للسطور الأخرى

كود PHP:

```
int main()
{
    printf("Hello world!\n");
    return 0;
}
```

ما نراه الآن هو عبارة عن ما يسمى ب **une fonction**
تقوم بجمع عدة أوامر للحاسوب . هذه **fonction** بصفة عامة
الأوامر تطلب من الحاسوب فعل شيئ محدد و واضح .
لها حدود فهي تبدأ بلامة{ و تنتهي بها } و إذا **fonction** ال
كنتم معي فإن هذه الفونكسيون.
السطر الأول و يتضمن

كود PHP:

```
    printf("Hello world!\n");
```

هذا السطر سنتحدث عنه في بقية الدرس .

كود PHP:

```
    return 0;
```

و هو الأمر أو الأوامر التي **instructions** هذا السطر يسمى ب
تجمعها الفونكسيون .

تنتهي دائما ب" ; " **instructions** ملاحظة جد مهمة : كل
النقطة الفاصلة هذا هو الخطأ الشائع الذي يقع فيه الكثيرون و
الذي وقعت فيه أنا أيضا.

```
#include <stdio.h>
#include <stdlib.h>
```
Directives de préprocesseur

```
int main(int argc, char *argv[])
{

  system("PAUSE");
  return 0;
}
```
Instructions

Fonction

كتابة شيئ على الشاشة

مثلا نريد كتابة WwW.BramjNeT.CoM is the BesT

تعطي للحاسوب أمر الكتابة و instructions ما علينا إلا كتابة هذا الأمر هو: printf

بصفة عامة يكتب هذا الأمر على الشكل التالي :

كود PHP:

```
printf("ما نريد كتابته ");
```

و بالتالي يصبح لدينا هذا البرنامج الذي يقول لنا WwW.BramjNeT.CoM is the BesT كالتالي :

كود PHP:

```
#include <stdio.h>
#include <stdlib.h>

int main(int argc, char *argv[])
{
  printf("WwW.BramjNeT.CoM is the BesT");
  system("PAUSE");
  return 0;
}
```

نجرب هذا البرنامج فيعطينا التالي :

C:\Program Files\CodeBlocks\console_runner.exe

WwW.BramjNeT.CoM is the BesTAppuyez sur une touche pour continuer...

BramjNeT.CoM The KinGSofT

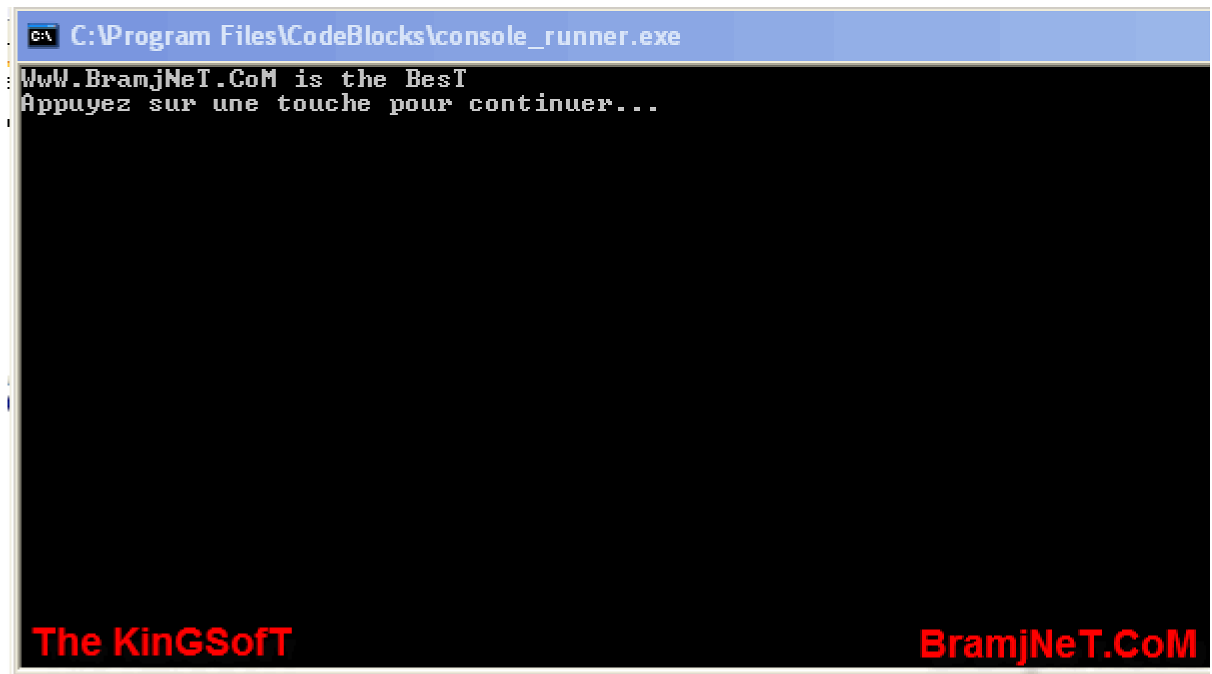
هناك ملاحظة واحدة لاحظها الكل و هي أن البرنامج عندما يكتب شيئا لا يعود إلى السطر لماذا هل هناك خطأ ؟؟؟ لا لا لن تقلقوا هناك بعد الكلمات التي ستساعدنا و هي :

\n : و هي للعودة إلى السطر (entrée).
\t : و هي تساعدنا في الجدولة .

إذا سيصبح الكود كالتالي :
كود PHP:

```
printf("WwW.BramjNeT.CoM is the BesT \n");
```

بعد التعديل



التعليقات (les commentaires).

التعليق و هو التعليق مثل اي تعليق في أي لغة أخرى و هو يستعمل إما للشرح أو للتفاهم بين المبرمجين .مثل التعليق في HTML...

لكتابة تعليق هناك عدة طرق و التي تتعلق بمدى طول التعليق :

+لكتابة تعليق قصير(في سطر واحد) نستعمل الرمز التالي (//)

كود PHP:

```
Code : C
// هذا تعليق
```

+لكتابة تعليق يحتوي على أكثر من سطر نستعمل الرموز التالية : (/*) (*/) و
لبداية التعليق (/*)-
لإنهاء التعليق (*/)-

كود PHP:

```
/* هذا
تعليق
يتكون من عدة أسطر
BramJneT
```

The KinGSofT */

و ننهي هذا الدرس بهذه المقولة من شركة IBM:
"Si après avoir lu uniquement les commentaires d'un programme vous n'en comprenez pas le fonctionnement, jetez le tout !"

مفاده : إذا قرأتم تعليقات برنامج فقط و لم تفهموا شيئا , فألقوا به !

أسئلة الدرس

1/ directive de préprocesseur هو سطر يبتدأ ب:
- #
- {
- //

2/ما هو الرمز الذي تنتهي به ال instruction؟
- /*
- ;
- }

3/ما هو إسم ال fonction ؟ التي نكتب بها نصا على الشاشة
- printf
- print
- pinrft

4/ما هو الرمز الذي يساعدنا على الرجوع إلى السطر ؟

- \t
- \n
- يكفي أن نضغط على entrée , أيها الأحمق البليد !!!.

5/تغليق من سطر واحد يبدأ ب :

- /*
- */
- //

إنتظروا الدرس القادم تحت

عنوان : عالم من المتغيرات les variables

أشهد أن لا إلـه إلا **الـلـه** و أن **محمدا** رسول **الـلـه**

نحن الآن في جزء من الدروس المهمة في البرمجة ألا و هي المتغيرات ...فأرجو من الجميع التركيز معي لأن الدروس القادمة ستعتمد بشكل كبير على هذا الدرس.

تلخيص و تذكير لما درسناه سابقا:
لقد تعلمتم في الدرس السابق كيف تصنعون مشروعا جديدا و كيف تنشؤون برنامجا بسيطا بالكونصول و ذلك بإستخدام ال IDE مثل Dev و Code::Blocks إلخ...
و قد شرحت لكم كم أنه من الصعب علينا نحن المبيدئين إنشاء برامج بالنوافذ لذلك و طيلة دروسنا القادمة سنتعامل فقط مع الكونصول.

أنتم الآن تعرفون كيفية كتابة نص على الشاشة .ممتاز.
قد تقولون أن هذا غير مهم , و السبب في ذلك أنكم لا تعلمون ما هي المتغيرات في البرمجة .
آه !! المتغيرات , كل لغات البرمجة فيها متغيرا كائنة من كانت و لغة السي ليست خارجة المجموعة .

**قد يتساءل البعض : و ما هو التغير ؟**

هذا الدرس له هدف للإجابة عن هذا السؤال . فأنا لا أريد أن أفقدكم الشوق لمعرفته . و لكن إعلموا أننا سنتحدث عن الأرقم , عن قيم يمكننا تخزينها في الذاكرة .

عنوان درسنا :

تعريف المتغير
إنشاء متغير
إظهار محتوى متغير
إسترداد قيمة مخزنة

تعريف المتغير

المتغير و بكل إختصاؤ هو عبارة عن قيمة أو معلومة صغيرة يتم تخزينها على الذاكرة Ram.
و سبب تسميته "متغير" هي أن قيمته يمكن أن تتغير خلال عمل البرنامج . مثلا , قيمة المتغير 8 هذه القيمة مثلا نعترها عدد حياة اللاعب في لعبة ما, هذه القيمة يمكنها أن تتغير كلما خسر اللاعب فتصبح 7 ...6 ...5 ....
برامجنا القادمة و كما أشرت سابقا ستكون معمرة بالمتغيرات لذلك أعطوني كل إنتباهكم .

في لغة سي , المتغير له شيئين رئيسيين :

* له قيمة: وهي العدد الذي يأخذه المتغير مثلا 6.
* له إسم : و هذا ما يساعدنا على معرفته .عندما نبرمج بلغة سي يجب علينا تحديد أسماء للمتغيرات .

كيفية تسمية متغير:

في لغة سي إذا كل متغير يجب أن يكون له إسم . و مثلا نريد تسمية متغير في لعبة و هو يمثل الوقت المتبقي لذلك نسميه مثلا :"Times" لكن هناك بعض القواعد التي يجب إحترامها:

1/تسمية المتغير لا تقبل سوى الأرقام و الحروف الكبيرة و الصغيرة :abAB012... مثلا.
2/إسم المتغير يجب أن يبدأ بحرف.
3/الفراغات ممنوعة منعا كليا , يكمننا إستعمال "underscore"

وهو الركز الوحيد المحل إستخدامه كرمز دون الجروف و _ الأرقام .
4/ليس لكم الحق في إستخدام les accents و هم : éàê ...

و أخيرا و هو الشيئ الأكثر أهمية , و يجبمعرفته و هو أن لغة سي تفرق بين الحروف الكبيرة و الصغيرة .. مثلا هذا المتغيران ليسا متشابهين :"Times", "times" .
و لكي تعلموا فلكل مبرمج طريقته الخاصة في تسمية المتغيرات .. لذلك أنصحكم بأن تبتكروا طريقة خاصة بكم في التسمية .

**مع أن لنا كل الحرية في إختيار إسم المتغير فهذا لا يعني تسمية المتغير عشوائيا بدون تفكير , بتعبير آخر يجب على المتغير أن يحمل إسم وظيفته -كنصيحة-.**

و الآن سنمر إلى ما هو أهم و هو :

أنواع المتغيرات:

حاسوبنا,كما لاحظتم و تلاحظون ليس, سوى آلة للحسابة فهو لا يعرف سوى معالجة الأرقام .
إليكم الآن مجموعة من أنواع الأرقام و التي تعرفونها بلا شك :

لدينا , الأعداد الحقيقية الموجبة :

45
357
14025

و لدينا الأعداد العشرية و هي تلك التي فيها الفاصلة :

75,909
1,7741
9810,7

إضافة إلى ذلك لدينا الأعداد الحقيقية السالبة ك

87-
128-

و أيضا أعداد عشرية سالبة :

76,12-

100,7589-

الحاسوب أو الكومبيوتر بحاجة لمساعدة ! فعندما نطلب منه تخزين عدد من الأعداد , يجب عليكم ذكر نوع هذا العدد ..

إذا عند إنشاء متغير يجب عليكم ذكر نوعه .
و إليكم جدولا يلخص الأنواع و قدراتها الإستعابية في لغة سي :

char	-128 à 127
int	-32 768 à 32 767
long	-2 147 483 648 à 2 147 483 647
float	-3.4 x 10 puissance 38 à 3.4 x 10 puissance 38
double	-1.7 x 10 puissance 308 à 1.7 x 10 puissance 308

الأنواع الثلاثة الأوائل يمكنون المبرمج من تخزين أعداد حقيقية . التوعين الأخيرين يمكنان من تخزين أعداد عشرية .

ستلاحظون أننا في مشوارنا الدراسي سنستخدم بشكل كبير الأعداد الحقيقية char, long و double.

إحذروا عند التعامل مع الأعداد العشرية !! فالكمبيوتر لا يعرف الفاصلة لذلك وجب إستخدام النقطة بدل كتابة الفاصلة .

بالنسبة للأنواع فأنا لم أذكرها كلها بل ركزت على الأساسيات .

إنشاء متغير

الآن أنشئوا مشروعا جديدا و أسموه "variables".
سننشئ الآن متغيرا أي سنطلب من الكومبيوتر أن يسمح لنا بإستخدام قليل من الذاكرة.

إنشاء متغير جد سهل الآن خاصة بعد أن تعلمنا الأمور الأساسية , فقط يجب علينا أن نقون بالآتي بالتتابع:

+ تحديد نوع المتغير .

+فراغ
+كتابة إسم المتغير
+و أخيرا لا تنسوا النقطة الفاصلة .

و بذلك نكتب كالآتي :
كود PHP:

```
long NameOfVariable;
```

هذا كل شيئ .

أظن أنكم فهمتم المبدأ.

هذه الحركة تسمى بإستدعاء المتغير **déclaration de variable.**
و نحن الآن . **fonction** و نقوم بهذه العميلة في أول أو بداية ال **fonction main..** و واحدة ألا و هي **fonction** نتعامل فقط مع
بذلك تكون عملية إستدعاء البرنامج تتم على الشكل التالي :

كود PHP:

```
#include <stdio.h>
#include <stdlib.h>

int main(int argc, char *argv[])
{
  // بداية ال fonction
}
  long BraKing;

  system("PAUSE");
  return 0;
  // نهاية ال fonction
}
```

عند عمل البرنامج تلاحظون أنه...لا يقوم بأي شيئ

شروحات لهذه الحالة :

في الحقيقة هناك أشياء تحدث لكن لا ترونها , فعندما يصل البرنامج إلى خانة الإستدعاء يطلب من الكومبيوتر بلطف: "هل يمكنني يا سيد كمبيوتر أن أقوم بإستغلال قليل من الذاكرة "
فيجيب الكومبيوتر: "بالطبع يا صديقي البيت بيتك "

للإختصار فمعظم المبرمجين عندما يريدون إنشاء مجموعة من المتغيرات مثلا 6 متغيرات , فبدل تضييع الوقت في الكتابة و إعادة الكتابة نقوم بالتالي :
كود PHP:

```
long Time, Level, Pro;
```

هذا الكود سينشأ 3 متغيرا من نفس النوع long و هي : Time و Level و Pro.
و الآن؟
بعد أن أنشأنا متغير يمكننا أن نعطي له قيمته أليس كذلك ؟؟

إعطاء قيمة للمتغير :

نريد مثلا أن نعطي للمتغير Level القيمة 3 .
هذا سهل جدااااااااااا يكفي أن نقوم بالكتابة كالتالي :

كود PHP:

```
Level = 3;
```

ليس هناك شيئ صعب لفعله فقط علينا إستدعاء المتغير ثم فراغ ثم علامة تساوي ثم نكتب قيمة المتغير ..و لا ننسى النقطة الفاصلة كالعادة.
الآن سنقوم بالتطبيق ... نعطي للمتغير Level القيمة 5 داخل برنامج :

كود PHP:

```
#include <stdio.h>
#include <stdlib.h>

int main(int argc, char *argv[])
{
  long Level;
  Level = 5;

  system("PAUSE");
  return 0;
}
```

هنا أيضا لا يحدث شيئ , الكل يحدث بين الذاكرة و البرنامج .

و كما قلت عن سبب تسميت المتغير "متغيرا" ..يمكننا اللعب و تبديل قيمة المتغير إلى أي قيمة و في نفس البرنامج :

كود PHP:

```
long Level;
Level = 5;
Level = 4;
Level = 150;
```

سيأخذ أولا القيمة 5 ثم بعد ذلك Level في هذا المثال المتغير القيمة 4 و القيمة 150 . بما أن الكومبيوتر سريع فإننا لا نشعر

بتغير قيمة المتغير... و ها قد إنتهى البرنامج

قيمة متغير

إليكم سؤالا مهما , و إذا كنت قد طرحته مسبقا فهذا دليل على أنك قد تصبح مبرمجا جيدا :

عندما نستدعي المتغير أية قيمة يأخذها ؟

في الحقيقة الحاسوب عندما يقرأ هذا السطر :

كود PHP:

```
long level;
```

يقوم بحجز مكان صغير في الذاكرة . و لكن ما هي قيمة المتغير في هذه اللحظة ؟هل توجد قيمة أصلية (0 مثلا).

إذا ... الجواب هو لا و لا و لا . لا توجد فيمة أصلية (par defaut).المكان محجوز لكن القيمة لا تتغير , أي أن المتغير يمكن أن يأخذ أية قيمة مهما كانت .
لذلك يجب الحذر كي لا نقع في مثل هذه الأخطاء في المستقبل و الأحسن دائما أن نعطي لمتغيراتنا قيما عندما نستدعيها.و في الأخير , هذا يتطلب منا دمج الإستدعاء و إعطاء القيمة :

كود PHP:

```
long Level = 5;
```

المتغير , هنا Level يأخذ القيمة 5.
الإيجابي هنا هو أننا نعرف أن متغيرنا يأخذ القيمة 5 و ليس أية قيمة مهما كانت .

الثوابت constantes

قد يأتي وقت ما ,نحتاج فيه إلى متغير ذو قيمة ثابتة لا تتغير طيلة فترة البرنامج .بمعنى آخر عندما نستدعي المتغير نريده أن تبقى قيمته ذابتة و ليس لأي أحد الحق في تغييرها .

نظرا لأن, constantes هذه المتغيرات الخاصة نسميها بالثوابت قيمتهم ثابتة لا تتغير طيلة عمل البرنامج .

مثال لإستدعاء ثابت:
كود PHP:

```
const long LEVEL_CONSTANT = 5;
```

ليس إجباريا, لكن إعتاد عمالقة سي على كتابة الثوابت بالحروف الكبيرة كما فعلت الآن . هذا يمكننا من تمييز الثوابت و المتغيرات بسهولة تامة. لاحظوا أنني إستعملت _ مكان الفراغ .

بدون هذا,فالثوابت لا تختلف في أي شيئ مع المتغير العادي . يمكنكم إعطاءها و غير ذلك مما سنتعلمه الآن .الشيئ الوحيد الذي يختلف, هو أننا عندما نحاول تغيير قيمة الثابت يغبرنا المترجم **compilateur** أن هناك خطأ في البرنامج.

أخطاء الترجمة تظهر أسفل البرنامج .. في هذه الحالة , المترجم سيظهر لكم كلمات كالآتي :
كود:

```
[Warning] assignment of read-only variable '
NOMBRE_DE_VIES_INITIALES'
```

إظهار محتوى متغير

حتى الآن تعلمنا كيف نظهر كتابة على الشاشة بإستعمال الدالة **printf** .
الآن سنتعلم كيفية إظهار قيمة(محتوى) بإستعمال نفس الدالة .

بنفس الطريقة, في الحقيقة نحن نستعمل الدالة **printf** بإستثاء أننا نضيف رمزا في المكان الذي نريد لقيمة المتغير أن تظهر فيه .
مثلا:

كود PHP:

```
printf("Il vous reste %ld vies");
```

هذا "الرمز الخاص و المميز" الذي تحدثت عنه هو **%** متبوع بالحرفين **ld**. هذه الحروف و الرموز تبين ماذا نريد أن نظهر."**ld**"تعني أن الرقم هو عدد صحيح.
هناك عدة رموز لكن و لأننا لا أقول مبتدئين لكن فوق المبتدئين بدرجة سنركز على هذان الإثان :
عدد صحيح (مثل : 4, 5 , 6 ,150) : **%ld**+
عدد عشري (5.18, 18.3, 120.32589): **%lf**+

في الدروس القادمة سنتحدث عن عدة رموز أخرى في وقت

مناسب فلكل مقام مقال. في وقتنا الحالي , إعلموا أننا عندما نريد إظهار قيمة متغيرات صحيحة (char, int, long...),يجب إستعمال %ld, و بالنسبة لأرقام عشرية (float, double), يجب إستعمال %lf.

شارفنا على النهاية . لكن عندما نظهر محتوى المتغير يجب أن نبين للدالة printf أي متغير يجب إظهار قيمته .
لكي نفعل هذا, يجب ذكر إسم المتغير بعد المزدوجتان و ذلك بعد إضافة فاصلة , مثلا:
كود PHP:

```
printf("Il vous reste %ld vies", Vies);
```

%ld سيعوض بقيمة المتغير المذكور بعد الفاصلة .
لماذا لا نقوم بتجربة كل هذا في برنامج ؟

كود PHP:

```
#include <stdio.h>
#include <stdlib.h>

int main(int argc, char *argv[])
{
  long nombreDeVies = 5; // Au départ, le joueur a 5 vies

  printf("Vous avez %ld vies\n", nombreDeVies);
  printf("**** B A M ****\n"); // Là il se prend un grand coup s
ur la tête
  nombreDeVies = 4; // Il vient de perdre une vie !
  printf("Ah desole, il ne vous reste plus que %ld vies maintena
nt !\n\n", nombreDeVies);

  system("PAUSE");
  return 0;
}
```

يمكن أن يكون هذا البرنامج لعبة (لا يتطلب الأمر سوى قليل من الخيال العلمي)

شرح البرنامج :

1. في البداية اللاعب لدية خمس نقاط حياة, نظهر هذا بإستعمال الدالة printf.
2. ثم يحدث إنفجار قرب اللاعب .
3. و في الأخير يصبح عدد نقاط حياته 4 , و نظهر هذه القيمة بواسطة printf.

كيفية إظهار محتوى عدة متغيرات في نفس ال printf .

يمكن لكل شخص أحب أن يظهر محتوى عدة متغيرات في نفس الدالة printf.يتطلب الأمر أن تبينوا المكان و نوعية العدد : %ld أو %lf , ثم تكتبوا أسماء المتغيرات كما في السابق لكن و بما أنه عندنا عدة متغيرات فيجب ذكرها و لكن حسب ترتيب الكتابة :

كود PHP:

```
printf("Vous avez %ld vies et vous etes au niveau n°%ld", Vies,
Level);
```

يجب إحترام الترتيب ففي هذا البرنامج ال%ld الأول يأخذ قيمة المتغير الأول Vies و%ld الثاني يأخذ قيمة المتغير الثاني Level.

إسترداد قيمة مخزنة

المتغيرات الآن بدأت تصبح مهمة . سنتعلم الآن أن نطلب من المستخدم إدخال قيمة و سنقوم بتخزينها في متغير .عندما ستتعلمون هذه الخاصية يسيكون بمقدوركم إنجاز الكثير من الأشياء الجميلة .

لكي نطلب من المستخدم إدخال قيمة , سنستعمل دالة أخرى ألا و هي الدالة :scanf.
هذه الدالة تشبه كثيرا الدالة printf .يجب عليكم إدخال ld% أو lf% بين مزدوجتين "" لكي نبين إن كنا نريد من المستخدم إدخال عدد صحيح أو عشري . ثم بعد ذلك يجب أن ذكروا إسم المتغير بعد الفاصلة ثم إضافة الركز &.
مثلا:

كود PHP:

```
scanf("%ld", &age);
```

. بين المزدوجتين %lf أو ld% لا يجب وضع سوى
لكن يجب وضع الرمز & أمام إسم المتغير الذي سيتلقى القيمة .

بالنسبة للرمز & سأشرح في الدروس القادمة لماذا نضعه , فلا تستعجلوا.

أسئلة الدرس :

1/ بين هذه المتغيرات من هو المتغير الخاطئ:

-positionMeNuS
-largueurFenêtre
-age_du_capitaine

2/ من هو نوع المتغير الذي سيمكننا من حفظ هذا العدد: 76.8

-char
-long
-double
-int

3/ لنفترض أن لدي متغيرا "bank" ذو القيمة 65000004 ماذا سيظهر هذا السطر في الشاشة :

كود PHP:

```
printf("Vous avez %ld euros sur votre compte", compteEnBanque);
```

-Vous avez %ld euros sur votre compte
-Vous avez 6 500 000 euros sur votre compte
-Vous avez d euros sur votre compte, compteEnBanque

تمرين الدرس

من الآن فصاعدا سأقوم بوضع تمرين لكل درس بما أننا بدأنا في لغة سي فعلا و تمرين اليوم هو :

أكتب برنامجا يطلب من المستخدم عمره و شهر ميلاده و يوم ميلاده و سنة ميلاده و يظهرها على الشكل التالي :

مرحبا بك أيها المستخدم ,من خلال ما أدخلت فأنت ذو العمر 16 ولدت في 1/8/1976.
عد في العام المقبل لتتحرى عمرك .

و إنتهى درسنا .

إنتظروا الدرس القادم تحت عنوان : العمليات الحسابية في لغة سي .

أشهد أن لا إلـه إلا **اللـه** و أن **محمدا** رسول **اللـه**

كما سبق و أشرت إليه في الدروس السابقة فحاسوبكم ليس سوى آلة للحساب . سواء كنتم تستمعون لموسيقى أو تشاهدون فيلما أو تلعبون لعبة , فالحاسوب لا يقوم سوى بعمليات حسابية منطقية .

في هذا الدرس سنتعلم كيفية القيام بمختلف العمليات التي يعرفها حاسوبكم , و سنستعمل ما تعلمناه في الدرس السابق ألا و هو المتغيرات . الفكرة و المحور الرئيسي الذي تدور عليه هو جمع المتغيرات "C كلمة "العمليات الحسابية في لغة ببعضها, ضربها , تخزين الناتج في متغير آخر ...إلخ.

مهما كان مستواك في الرياضيات فهذا الدرس لا يسمح بتجاوزه أبدا.فلنتكلم بصراحة و صدق : إذا لم تكن تعرف الجمع , إذا فأنت غير مؤهل لتعلم أي لغة برمجية .

عنوان درسنا:

العمليات القاعدية
الإختصارات
المكتبة الرياضية La librairie

mathématique

العمليات القاعدية :

الشيئ الذي يجب معرفته هو أن الحاسوب هو عبارة عن آلة حاسبة متمكنة و ممتازة , طالما هو يعرف العمليات البسيطة:

الجمع *
الضرب *
الطرح *
القسمة *
Modulo (سأقوم بشرحه فيما بعد) *

إذا أردتم القيام بعمليات معقدة (الأس , الخوارزميات , الجذور المربعة...) يجب عليكم برمجتهم أي يجب تعريفهم للحاسوب . من حسن حظنا , وهذا ما سنراه في بقية الدرس , أن هناك مكتبة رياضية تضم مجموعة من الدوال الرياضية الجاهزة. إذا فلن تحتاجوا لإعادة كتابتهم , إلا إذا كنتم أساتذة رياضيين فهذا الحل ممكن أيض ا.

لنبدأ الآن بعملية الجمع.
لكي نقوم بعملية الجمع , سنستعمل الرمز +(إنني لا أمزح) و يجب عليكم وضع الناتج في متغير آخر . إذا نسقوم مثلا و Long أي ناتج من نوع "Resultat" بإستدعاء متغير و لنسمه يكون الناتج على هذا الشكل:

كود PHP:
```
long Resultat = 0;

Resultat = 5 + 3;
```

لا يتطلب منك الأمر أن تكون محترفا لكي تخمن أن المتغير "Resultat" سيأخذ القيمة 8 أثناء التشغيل .

بالطبع لا شيئ يظهر في الشاشة مع هذا الكود .لكن إذا أردتم كما تعلمت أن تفعلوا Printf رؤيت قيمة المتغير , أضيفوا الدالة في الدرس السابق :

كود PHP:
```
printf("5 + 3 =  %ld", Resultat);
```

هذا هو الجمع الذي يعرفه الحاسوب.
بالنسبة للعمليات الأخرى, فمثل سابقتها, فقط الرمز الذي يتغير :

* الجمع: +
* الطرح: -
* الضرب: *
* القسمة: /
* Mdulo:%

إذا كنتم قد إستعملتم الآلة الحاسبة في جهازكم فستعرفون هذه الرموز حق معرفتها .
ليس هناك أية صعوبة فيما يخص العمليات, بإستثناء الإثنين الأخيرين (القسمة و Modulo). لذلك خصصت لهم فقرة لنتحدث عن كل واحدة بدقة .

القسمة

القسمة تتم على الحاسوب جيدا عندما لا يكون عندنا الباقي.
مثلا, 6 / 3 تساوي 2, فالحاسوب هنا يعطيكم الخارج الصحيح, إلى هنا ليس هناك مشكل.

لنأخذ الآن مثالا آخر لقسمة لها باقي مثل : 5 / 2.
تعطينا إذا قمنا بحسابها جيدا, تساوي 2.5 ,5 / 2

و مع ذلك أنظروا لهذا البرنامج ماذا يفعل:

كود PHP:

```
long resultat = 0;

resultat = 5 / 2;
printf ("5 / 2 = %ld", resultat);
```

هناك مشكل كبير. طلبنا من الحاسوب أن يقوم ب 2 / 5, فكنا ننتظر منه أن يعطينا 2.5 كنتيجة, لكن هذه الآلة الحاسبة المتمكنة و الممتازة أعطتنا 2!!!!

ما هذا؟؟؟ هل أجهزتنا غبية لهذا الحد ؟
في الحقيقة, عندما يرى الأعداد 5 و 2, يقوم الحاسوب بقسمة لأعداد صحيحة. هذا يعني أنه سيخطئ في النتيجة لأنه لا يحتفظ إلا بالطرف الصحيح(2).

آه أستاذ..استاذ أنا أعرف ! هذا لأننا إستعملنا المتغير Resultat من نوع **long** ! لو كان من النوع **double**, لإستطعنا أن نخزن

قيمة عشرية !!!

و مع ذلك ,

جربوا نفس الكود السابق مع تغيير نوع المتغير الذي نخزن فيه النتيجة إلى النوع double, و سترون أن النتيجة لا تتغير.

إذا أردنا من الحاسوب أن يعطينا النتيجة الصحيحة, يجب علينا تغيير أعداد العملية 5 و 2 إلى أعداد عشرية, أي أن نكتب 5.0 و 2.0 (إنها نفس الأعداد السابقة, لكن بالنسبة للحاسوب فهي أعداد عشرية, لذلك فهو يقوم بقسمة لأعداد عشرية) :

كود PHP:

```
double resultat = 0;

resultat = 5.0 / 2.0;
printf ("5 / 2 = %lf", resultat);
```

العدد الناتج الآن صحيح.

لكن ستسألونني, كيف يمكننا معرفة الباقي في القسمة؟
هنا يتدخل الخارق: Modulo .

MODULO

الموديلو هي عملية رياضية تمكننا من الحصول على باقي قسمة معينة. قد يكون عملية حسابية غير معروفة مثل الأربعة الآخرين, و لكن بالنسبة للحاسوب فهي عملية قاعدية...

Modulo, كما قلت يمثل بالرمز %.
إليكم بعض الأمثلة :

5 % 2 = 1
14 % 3 = 2
4 % 2 = 0

إنتهى, هذا ما لدي بالنسبة للموديلو ليس هناك أي شيئ آخر. فقط قمت بتعريفه للذين لا يعرفونه.

العمليات بين المتغيرات :

الشيئ المهم الآن, بعد أن تعرفتم إلى العمليات الخمس القاعدية , سنقوم الآن بالتدرب على بعض العمليات الأكثر

أهمية.
في الحقيقة لا شيئ يمنعنا من فعل :

كود PHP:
```
resultat = nombre1 + nombre2;
```

هذا السطر يقوم بجمع قيمتي المتغيرين nombre1 و nombre2, و يخزن الناتج في متغير آخر "resultat".

آآآآه هنا الأشياء بدأت تصبح مهمة و مهمة جدا
http://www.siteduzero.com/Templates/...lies/smile.png

أتت لي فكرة رائعة. لديكم الآن المستوى للقيام بمحسبة صغيرة نعم نعم , أأكد لكم ذلك. !

تخيلوا معي برنامجا يطلب من المستخدم رقمين . هذه الأرقام, نخزنها في متغيرات.
ثم, نقوم بجمع هذه المتغيرات, و نخزن الناتج في متغير آخر و لنسمه "resultat".

و من ثم لا عليكم سوى إظهار الناتج على الشاشة.

حاولوا أن تقوموا بعمل هذا البرنامج بأنفسكم, هذا سهل سيدريكم.
الجواب في الأسفل قليلا:
.
.
.
.
.
.
كود PHP:
```
int main(int argc, char *argv[])
{
  long resultat = 0, nombre1 = 0, nombre2 = 0;

  // نطلب من المستخدم إدخال رقمين :

  printf("Entrez le nombre 1 : ");
  scanf("%ld", &nombre1);
  printf("Entrez le nombre 2 : ");
  scanf("%ld", &nombre2);

  // نقوم بالعملية:

  resultat = nombre1 + nombre2;
```

```
  // نظهر ناتج الجمع:

  printf ("%ld + %ld = %ld\n", nombre1, nombre2, resultat);

  system("PAUSE");
  return 0;
}
```

هنا قمنا بعمل أول برنامج مهم حتى الآن ...أليس كذلك ؟

أنصحكم بأن تقوموا بنفس العملية مع العمليات الأخرى...
يمكنكم أيضا إضافة متغير آخر ليقوم الحاسوب بجمع متغيرات ثلاث و الطريقة تعمل 100%:

كود PHP:

```
resultat = nombre1 + nombre2 + nombre3;
```

الإختصارات:

كما سبق و أشرت فقد قمنا بدراسة جل العمليات ... فبهذه العمليات الخمس يمكنكم عمل كل شيئ .أنا أعرف أنه من الصعب تصديق أن لعبة ثلاثية الأبعاد لا تقوم إلا على عمليات بسيطة مثل + و - و ... و لكنها الحقيقة المرة .

تقنيات تمكننا من إختصار كتابة العمليات C يوجد في لغة.
و لماذا نستعمل الإختصارات؟لأننا, غالبا, نستعمل عمليات متكررة. ستفهمون ما أعنيه و ذلك بعد أن تتعرفوا على ما نسميه ب:incrémentation .

incrémentation

في غالب الأحيان ستحتاجون إلى زيادة 1 إلى المتغير. مهما كان نوع البرنامج, ستكون هناك متغيرات تزداد قيمتها واحدا بواحد.
هل تستطيعون إضافة 1 على "N". تخيلوا معي متغيرا إسمه المتغير, دون معرفة القيمة التي يحتويها؟
إليكم الطريقة :

كود PHP:

```
N = N + 1;
```

أجل, !!! N ثم نخزن الناتج في N+1 ماذا هناك؟ نقوم بحساب فهي الآن 5 و إذا كانت 8 فهي الآن 4 N إذا كانت قيمة المتغير

9 ...

تعتبر هذه العملية تكرارية. المتخصصون في هذا المجال لن يتركونا كالحمقى نكتب إسم المتغير مرتين.
و هكذا قاموا بعمل إختصار لهذه الكتابة التي نسميها incrémentation. الكود الذي في الأسفل يفعل نفس الشيئ كالذي في الأعلى :

كود PHP:
```
N++;
```

هذه الطريقة فعالة خصوصا في; المستقبل فكما قلت سنكون مضطرين إلى إضافة 1 إلى المتغير.
كما لاحظتم لدينا الرمز : ++ الذي يوجد أيضا في إسم لغة البرمجة المشهورة : ++C. أظنكم فهمتم من أين أتى هذا الإسم العجيب فهو يعني لغة C (+1) معدلة و مطورة.

décrémentation

و هي عكس زيادة 1 أي : ننقص 1 من قيمة المتغير. النقصان عندما نكتبه بصيغته الكاملة :

كود PHP:
```
N = N - 1;
```

و بصيغته المختصرة :

كود PHP:
```
N--;
```

إختصارات أخرى

يوجد العديد من الإختصارات و التي تعمل على نفس المبدأ السابق. لكن هذه المرة هذه الإختصارات تعمل على جميع العمليات : + - * / %.
و تمكننا الإختصارات من التخلص من تكرار إسم المتغير المراد إجراء العملية عليه عدة مرات. كذلك إذا أردتم ضرب متغير في 2 :

كود PHP:
```
N = N * 2;
```

و يمكنكم كتابتها بطريقة مختصرة :

كود PHP:

```
N *= 2;
```

إذا كان المتغير يحمل القيمة 12 فسيحمل القيمة 24 عند نهاية الأمر.
بالنسبة لباقي العمليات فهي تقام بنفس الطريقة. إليكم برنامجا مصغرا كمثال :

كود PHP:

```
long N = 2;

N += 4; // N = 6...
N -= 3; // ... N = 3
N *= 5; // ... N = 15
N /= 3; // ... N = 5
N %= 3; // ... N= 2
```

هيا!!!!!! فكروا قليلا, فالتفكير لم يقتل أحدا يوما.
ما يجب تذكره هو أن الزيادة ب1 يبقى الإختصار الأكثر إستعمالا.

المكتبة الرياضية La librairie mathématique

في لغة سي يوجد ما يسما بالمكتبات و المكتبات هي عبارة عن مجموعة من الدوال الجاهزة. هذه الدوال قد كتبت من طرف مبرمجين قبلكم و هي تساعدكم في عدم تكرار العمل الذي قاموا به.
لقد إستعملتم الدالتين printf & scanf من المكتبة stdio.h.
و ما يجب معرفته هو أن هناك مكتبة أخرى تسمى math.h, و التي تحتوي على العديد من الدوال الرياضية الجاهزة.

لإستعمال هذه المكتبة التي تحتوي على الدوال مفيدة من المهم عدم نسيان وضع هذا الأمر في أعلى البرنامج:

كود PHP:

```
#include <math.h>
```

نأتي الآن إلى تقديم أهم الدوال-سأقوم فقط بذكر أهمها و أكثرها إستعمالا-.

fabs

تقوم هذه الدالة بإعادة القيمة المطلقة لعدد, أي |x|.
بإختصار فهي تعيد لنا(تقوم بالإرجاع بعد إنجاز مهمتها) القيمة الموجبة للعدد المدخل:

كود PHP:
```
double absolu = 0, N=-27;

absolu = fabs(N); // absolu = 27
```

إذا فالمتغير "absolu" هذه الدالة تعيد متغيرا من النوع double يجب ان يكون من نفس النوع كذلك... هناك دالة أخرى تشبهها تسمى "abs" و الإختلاف في نوع المتغير الذي تعيده كل واحدة على حدة.

pow

هذه الدالة تقوم بحساب أس عدد ما. يجب تعريف قيمتين: العدد و الأس. هكذا:

كود PHP:
```
pow(العدد, الأس(القوة));
```

sqrt

هذه الدالة تقوم بحساب الجذر المربع لعدد معين. تعيدقيمة من نوع double double.

كود PHP:
```
double resultat = 0, N = 100;

resultat = sqrt(N); // resultat = 10
```

sin, cos, tan

يقوم هؤلاء الثلاثة بحساب جيب و جيب تمام و ظل زاوية ما و يجب أن تكون القيمة المعطاة بالراديان.

هناك العديد من الدوال الأخرى لكن لكبر مستواها في الرياضيات لم أرد ذكرها فهي لا تستعمل كثيرا ...

إنتهى درسنا أصدقائي صديقاتي و الآن مع التمرين المخصص لهذا الدرس .

نص التمرين :

قم بعمل برنامج يطلب من المستخدم إدخال أربعة أرقام و يقوم بالتالي :

- يظهر جمع و طرح و قسمة و ضرب الأعداد الأربعة .
- يظهر الجذر المربع للجمع و الضرب و الطرح و القسمة
- يقوم بجمع الجذور المربعة و يقوم بكتابتها و سط الشاشة .

هذا التمرين له هدف بتدريبكم على القيام بمعظم العمليات فهو يساعدكم على أن تصبحوا أكثر ليونة ... أنتظر إبداعاتكم إخواني ....

ترسل الإجابات على الخاص و يوضع الحل في الشهر القادم إن شاء الله ...

إنتظر الدرس القادم تحت عنوان : الإحتمالات Les conditions

أشهد أن لا إلـه إلا **الـلـه** و أن **محمدا** رسول **الـلـه**

رأينا في الدرس السابق أنه يوجد عدة لغات برمجية. البعض عدا أنه يصلح C, قريبة جدا من PHP منهم يشبه الآخر: فمثلا لغة للقيام بمواقع و ليس ببرامج

وجدت منذ القديم, مما يجعل منها نموذجا C في الحقيقة لغة للغات أخرى جديدة.
أغلبية لغات البرمجة لديهم في آخر المطاف صفات متشابهة, تتشابه في الأساسيات.

بالتحدث عن الأساسيات : نحن الآن داخل الخليط. رأينا كيفية إنشاء متغيرات, القيام بعمليات بإستعمالهم, سننتقل الآن إلى الإحتمالات.
بدون إحتمالات, برامجنا ستقوم دائما بنفس العمل, مما سيضفي عليها الكثير من الملل

الإحتمال "if... else"
booléens, قلب الإحتمالات
الإحتمال "switch"

الإحتمال "if... else"

تقوم الإحتمالات بمقارنة المتغيرات. فيمكننا أن نقول"إذا كان يساوي القيمة 50, قم بهذا"...و لكن سيكون من i المتغير الممل أن لا نقوم بمقانة سوى التساوي ! يجب أيضا أن نقارن إذا كان المتغير أصغر من 50 , أصغر من أو يساوي 50, أكبر من,

أكبر من أو يساوي ...

حضرت الكل C لا تقلقوا فلغة

سنتبع المخطط "if... else", لتعلم كيفية إستخدام الإحتمال التالي:

1/ بعض الرموز التي يجب معرفتها قبل البدأ.
2/ المقارن if
3/ المقارن else
4/ المقارن "else if"
5/ عدة إحتمالات في نفس الوقت

هناك على C, في "if... else" قبل البدأ بكتابة إحتمال من النوع 2-3 رموز قاعدية يجب معرفتها. هذه الرموز تعتبر مهمة للقيام بإحتمال ما .

بعض الرموز التي يجب معرفتها قبل البدأ

الرمز	معنـــاه
==	يساوي
>	أكبر قطعا
<	أصغر قطعا
>=	أكبر من أو يساوي
<=	أصغر من أو يساوي
!=	يخالف

**إنتبهوا جيدا, يوجد رمزان "==" لمقارنة التساوي. خطأ شائع يرتكبه المبتدؤون الذي يتجلى في وضع رمز واحد =, الذي ليس له نفس الدلالة في لغة C. سنتحدث عن ذلك في الأسفل.**

المقارن if

فلنهاجم الآن المهم بدون أي إنتظار 😊
سنقوم بإحتمال بسيط, و الذي سيقول للحاسوب :

كود PHP:

```
إذا كان المتغير يساوي قيمة معينة
إذن قم بالآتي
```

و هذا هو اللفظ المستعمل "if". بالإنجليزية, الكلمة "إذا" تترجم للقيام بإحتمال.
إفتحوا القوسين : داخل داخلهما يجب عليكم كتابة .if أكتبوا الشرط.

بعد ذلك إفتحوا { و أغلقوها بعيدا }. كل ما هو موجود داخل المزدوجتين سيتم تنفيذه فقط إذا تحقق الشرط الموجود داخل القوسين.

و نخلص بعد كل ذلك بهذا الكود:

كود PHP:

```
if(//شرطكم)
{
  // الأوامر المراد تنفيذها إذا تحقق الشرط
}
```

فلنضع مكان التعليق "شرطكم", سنكتب شرطا لكي نقارن متغيرا ما.
الذي يحتوي على عمركم. "age" مثلا, يمكننا مقارنة المتغير
الآن سنقوم بمعرفة هل أنت راشد , أي هل عمرك أكبر من 18 سنة...

كود PHP:

```
if (age >= 18)
{
  printf ("أنت راشد !");
}
```

الرمز => يعني أكبر من أو يساوي, كما رأيناه في الجدول أعلاه.

**إذا وجد أمر واحد بين {} فهي تصبح إختيارية فيمكننا كتابة:**

كود PHP:

```
if (age >= 18)
  printf ("أنت راشد !");
```

فلنجرب هذا الكود

إذا كنتم تريدون تجربة الأكواد السابقة و ذلك لمعرفة كيفية إشتغال if, يجب وضع ال if وسط الدالة main و لا ننسى أيضا إستدعاء المتغير age من إختيارنا قيمة و التي سنعطيها.
قد تبدوا هذه التجربة سخيفة في نظر البعض لكنها مهمة جدا فهي ستساعد على ترسيخ مفهوم الإحتمال و كيفية إشتغاله.

إليكم كودا كاملا يمكنكم تجريبة:

كود PHP:

```
#include <stdio.h>
#include <stdlib.h>

int main(int argc, char *argv[])
{
    long age = 20;

    if (age >= 18)
    {
      printf ("Vous etes majeur !\n");
    }

    system("PAUSE");
    return 0;
}
```

هنا المتغير age يساوي 20 إذن فسيتم القيام بالأمر الموجود داخل {} أي أن الرسالة "Vous êtes majeur !" ستظهر على الشاشة.
حاولوا الآن تغيير قيمة المتغير age غيروها مثلا للقيمة 10: . الشرط سيكون خاطئا و بالتالي الرسالة لن تظهر.

إعتمدوا على هذا الكود لتقارنوا الأكواد القادمة...

المقارن "else"

else تعني و إلا...
الآن نعرف كيفية القيام بمقارنة بسيطة, فلنذهب بعض الشيئ بعيدا: إذا لم تنجح المقارنة (الشرط خطأ), يجب أن نقول

للحاسوبنا أن يقوم بأمور أخرى instructions.

إذا سنكتب :

كود PHP:

```
إذا كان هذا المتغير== هذه القيمة
قم بهذا
و... إلا فقم بهذا
```

بعد: { و سيصبح لدينا else لا يجب علينا سوى إضافة

كود PHP:

```
if (age >= 18) // إذا كان العمر أكبر من أو يساوي
{
  printf ("أنت راشد !");
}
else // و... إلا
{
  printf ("أنت قاصر !");
}
```

أظن أن هذا الكود مفهوم للجميع...
إذا لم تفهم شيئا لا تترد في طرحه فنحن هنا لمساعدتك...

المقارن "else if"

else if إذا و إلا و تعني...

و تتيح لنا لغة سي أيضا جمعهما. if و else رأينا كيف نقوم ب
في تعبير واحد و هو else if...
و ذلك للقيام بمقارنة أخرى إذا لم يتحقق الشرط الأول.
else if توجد بين if و else

فنكتب:

كود PHP:

```
إذا كان هذا المتغير== هذه القيمة
قم بهذا

و إلا إذا كانت == هذه القيمة
قم بالآتي
```

و... إلا فقم بهذا

فيصبح لدينا كود على الشكل التالي:

كود PHP:

```
if (age >= 18) // إذا كان age 18 يساوي أو من أكبر
{
  printf ("أنت راشد !");
}
else if ( age > 4 ) // إلا إذا كان age 4 و من قطعا أكبر
{
  printf ("لا زلت صغيرا...");
}
else // و... إلا
{
  printf ("Aga gaa aga gaaa gaaa"); )
}
```

يقوم الحاسوب بالإختبارات بالترتيب:

1. الأول: إذا كان الشرط صحيحا يقوم if في الأول يقوم ب بالأمر الموجود داخل المعقوفتين الأوليتين.

2. و يقوم بإختبار "else if" إذا لم يتحقق الشرط الأول يذهب ل جديد: إذا كان الشرط صحيحا ينجز الأمر الموجود بين المعقوفات.

3. و أخيرا إذا لم يتحقق أي شرط من الشروط السابقة يقوم بأوامر "else"

"else" و "else if" ليسا إجباريين.للقيام بإحتمال لا نحتاج على أقل تقدير if سوى ل
فيمكننا كتابة "else if" لاحظوا أن بإمكاننا إستعمال الكثير من:

كود PHP:

```
if(...)
//الأمر1
else if(...)
//الأمر2
else if(...)
//الأمر3
else if(...)
//الأمر4
else if(...)
//الأمر5
else(...)
```

الأمر6

عدة إحتمالات

واحد. مثلا, if قد يكون أيضا القيام بعدة إحتمالات في داخل
تريدون أن تعرفوا إذا ما كان العمر أكبر من 18 و أصغر من 25 .

لهذا, يجب إستعمال رموز جديدة:

&&------> "و"
||------> "أو"
!------> "نفي"

&&

إذا كنا نريد أن نقوم بالإحتمال المذكور أعلاه, يجب كتابة:

كود PHP:

```
if (age > 18 && age < 25)
```

العلامتان && يعنيان "و". الإحتمال سيقرأ بالعربية:"إذا كان العمر أكبر من 18 و العمر أصغر من 25".

||

للقيام ب "أو" يجب إستعمال الرمزين ||.
و للقيام بهما فهناك عدة طرق و ذلك حسب نوعية لوحة المفاتيح, إذا ما عليك سوى تجريب إحدى الطرق:





تخيلوا معي برنامجا يقوم بتحديد إذا ما كان الشخص يستطيع فتح حساب بنكي .
و كما تعرفون لفتح حساب بنكي يجب أن لا تكون صغيرا و لنقل

مثلا لا يجب أن يقل عمرك عن 30 سنة أو أن يكون لديك الكثير من المال...
إحتمالنا لمعرفة هل الزيون قادر على فتح حساب بنكي أو لا سيكون كالتالي:

كود PHP:

```
if (age > 30 || money > 100000)
{
  printf("مرحبا بك في سوبر بنك !");
}
else
{
  printf("****أغرب عن وجهي أيها !");
}
```

!

هذا الرمز الأخير و الذي هو علامة تعجب يعني في البرمجة "نفي".
يجب وضع هذا الرمز قبل الشرط و ذلك ليكون المعنى: إذا لم هذا الشرط صحيحا.

كود PHP:

```
if (!(age < 18))
```

يمكن ترجمة الكود فوق بالتعبير التالي:
إذا لم يكن الشخص قاصرا.
لو حذفنا الرمز ! لكان معنى الترجمة كالتالي:
إذا كان الشخص قاصرا.

بعض الأخطاء التي يقع فيها الكثيرون

1/ نسيان العلامتين == ووضع واحدة مع أن معناهما مختلف
2/ زيادة النقطة الفاصلة بعد الإحتمال if.

أرجو من الجميع الإنتباه إلى هذه الأخطاء فهي ستسبب لكم الكثير من المشاكل في المستقبل.

Booléens,القلب النابض للإحتمالات.

سندخل الآن في المزيد من التفصيلات حول كيفية عمل الإحتمال if....else.
أضافت الإحتمالات شسئا يسمى ب booléen.

إنه شيئ جد مهم لذلك ففتحوا آذانكم جيدا(نسيت بل إفتحوا أعينكم.)

بعض الأمثلة للفهم

إليكم هذا الكود الذيسأطلب منكم تجريبه:
كود PHP:

```
if (1)
{
    printf("C'est vraiصحيح");
}
else
{
    printf(" خطأ");
}
```

**لكن ما مهنى هذا؟ لقد وضعت العدد فقط كشرط لهذا الإحتمال؟؟؟**

ستفهمون بعد قليل ...
الآن قوموا بعمل تجربة أخرى لكن بكود مختلف:

كود PHP:

```
if (0)
{
    printf("خطأ");
}
else
{
    printf("صحيح");
}
```

ستروون أن النتيجة هي عكس الأولى.
الآن قوموا بعدة تجارب و ذلك بإستبدال 0 بأي قيمة أخرى مثل:
36- ,10- ,226 ,15 ,4...

أي كلمة تظهر على الشاشة؟ نجد دائما "صحيح".

إستنتاج:إذا وضعنا الصفر الشرط يكون خاطئا و إذا وضعنا أي عدد آخر يخالف الصفر يكون الشرط صحيحا.

الشرح:

هذا الإحتمال يعيد if في الحقيقة ففي كل مرة تقومون بها ب لنا قيمة 1 إذا كان الشرط صحيحا و 0 إذا كان خاطئا.

مثال:

كود PHP:

```
if (age >= 18)
```

لنفترض أن age أو من أكبر 23 صحيح فالشرط إذن .23 يساوي بالعدد 1 age >= 18 يساوي 18 فيقوم الحاسوب بتغيير.
و كذلك بالمثل إذا كان الشرط خاطئا يعوضه ب 0.

التجربة بإستعمال متغير

فلنقم بالتجربة كيف نرسخ المفاهيم السابقة: قوموا بإرسال نتيجة الإحتمال إلى متغير, كما لو كانت عملية.

كود PHP:

```
long age = 20;
int majeur = 0;

majeur = age >= 18;
```

```
printf("Majeur = : %ld\n", majeur);
```

كما تلاحظون فإن قيمة المتغير majeur 1 هو .
مثلا هنا ستتغير القيمة age=10 قوموا بنفس العمل لكن بأخذ
إلى 0.

المتغير "majeur" يسمى booléen.

نسمي المتغير الذي يأخذ القيم 0 و 1 booléen.
كذلك:

خطأ=0
صحيح=1

لكي نكون دقيقين ف 0 يساوي خطأ و جميع الأعداد الأخرى
التي تخالف الصفر صحيحة.
لكن و كما جرت العادة لن نقوم سوى بإستعمال 0 أو 1 .

بالنسبة لنا نحن متعلمي لغة سي فلا يوجد نوع خاص لهذا
كنوع خاص .... و "bool" المتغير++ لكن تم إضافته في سي
هذا ما سنراه إن شاء الله.

للإختصار.bool من الآن و صاعدا سأقوم بتغيير booléen ب

لذلك فإن التوع المستعمل لل bool ك هو أي نوع: char-
double-long-int ...
bool و لا تهتموا بالنوع فقط يجب معرفة كيفية إستخدام
بطريقة حكيمة.

bool و الإحتمالات

كود PHP:

```
int majeur=1;

if (majeur)
{
  printf("أنت راشد !");
}
else
{
```

```
  printf("أنت قاصر");
}
```

يحمل القيمة 1 الشرط صحيح majeur بما أن المتغير .

إليكم إحتمالا آخر:
كود PHP:

```
if(majeur && man)
```

هذا الإحتمال يترجم كالتالي: إذا كنت راشدا و كنت ذكرا.
يبين إذا ما كان شيئ صحيح أو خطأ إذن bool.
ما قمنا به حتى الآن قد يبدو لا فائدة منه لكن بالتعمق في البرمجة نجد أن له إستعمالات كثيرة و مفيدة جداااالل...

الإحتمال Switch

الإحتمال if ...else الأكثر إستعمالا هو من قبل رأيناه الذي.
لكن في بعض الأحيان نجد أنفسنا نقوم بالتكرار كمثال الكود أسفله:

كود PHP:

```
if (age == 2)
{
  printf("Salut bebe !");
}
else if (age == 6)
{
  printf("Salut gamin !");
}
else if (age == 12)
{
  printf("Salut jeune !");
}
else if (age == 16)
{
  printf("Salut ado !");
}
else if (age == 18)
{
  printf("Salut adulte !");
}
else if (age == 68)
{
  printf("Salut papy !");
}
```

```
else
{
  printf("Je n'ai aucune phrase de prete pour ton age  ");
}
```

القيام ب switch

لتفادي القيام بعدة إحتمالات على متغير واحد, تم إبتكار طريقة إخرى للقيام بالإحتمالات.
هذه الطريقة تسمى switch و هذا مثال عليه:

كود PHP:

```
switch (age)
{
case 2:
  printf("Salut bebe !");
  break;
case 6:
  printf("Salut gamin !");
  break;
case 12:
  printf("Salut jeune !");
  break;
case 16:
  printf("Salut ado !");
  break;
case 18:
  printf("Salut adulte !");
  break;
case 68:
  printf("Salut papy !");
  break;
default:
  printf("Je n'ai aucune phrase de prete pour ton age  ");
  break;
}
```

نكتب:

switch(المتغير1)
و ذلك للدلالة على أنني سأقوم بعمل الإحتمالات على المتغير1.
نفتح المعقوفات و نغلقها تحت.

و داخل المعقوفات نتعامل مع الحالات الممكنة: case 2, case 4, case 5, case 45...

**بعد كل حالة و هو إجباري إذا لم يجب وضع الأمر break; تقوموا بذلك فالحاسوب سيقرأ جميع الحالات و يقوم بجميع الأوامر...**

لذلك حذاري من نسيان هذا الأمر.

و لا ننسى الأمر default الذي يعمل عمل else إذا لم يحقق المتغير1 الحالات المذكورة يقوم الحاسوب بالقيام بأوامر default.

القيام بقائمة للأطعمة :

يقوم السويتش عادة بالقيام بالقائمات في الكونصول. إذا إلى العمل.

المطلوب منكم:

أن تقوموا بعمل قائمة للأطعمة التي يقدمها مطعمكم و أن تعرضها على المستعمل فيختار منها ما يريد ثم تكتب لقد إخترت ما يلي... مثلا.

ماذا تنتظرون هيا للعمل

الحل:

كود PHP:

```
#include <stdio.h>
#include <stdlib.h>

int main(int argc, char *argv[])
{
  long choixMenu;

  printf("=== Menu ===\n\n");
  printf("1. Royal Cheese\n");
  printf("2. Mc Deluxe\n");
  printf("3. Mc Bacon\n");
  printf("4. Big Mac\n");
  printf("\nVotre choix ? ");
  scanf("%ld", &choixMenu);

  printf("\n");

  switch (choixMenu)
  {
    case 1:
       printf("Vous avez choisi le Royal Cheese. Bon choix !");
       break;
    case 2:
       printf("Vous avez choisi le Mc Deluxe. Berk, trop de sauc
e...");
```

```
            break;
         case 3:
            printf("Vous avez choisi le Mc Bacon. Bon, ca passe encor
e ca ;o)");
            break;
         case 4:
            printf("Vous avez choisi le Big Mac. Vous devez avoir tre
s faim !");
            break;
         default:
            printf("Vous n'avez pas rentre un nombre correct. Vous ne
mangerez rien du tout !");
            break;
    }

    printf("\n\n");

    system("PAUSE");
}
```

و هاقد إنتهينا...
في آخر السويتش **default** أظن أنكم لم تنسوا
في الحقيقة عندما تبرمجون يجب عليكم التفكير في جميع الإحتمالات.
ستجدون دائما من يضغط على الرقم 10 أو سيكتب مرحبا مع أنكم لم تنتظروا ذلك.

لذلك لا تضعوا ثقتكم في المستخدم يستطيع دائما إدخال أي شيئ لذلك يجب أخذ كل شيئ بعين الإعتبار.

بالنسبة للتمارين فسأضعها في درس خاص بها بما أنكم أصبحتم تتقدمون...

سامحوني على تأخر الدرسو ذلك لظروف شخصية
فسامحوني لقد وجدت اليوم متسعا من الوقت فكتبت ما تبقى من الدرس...

إنتهى
و بالتوفيق للجميع

أشهد أن لا إلـه إلا **الـلـه** و أن **محمدا** رسول **الـلـه**

رأينا في الدرس السابق أنه يوجد عدة لغات برمجية. البعض عدا أنه يصلح C, قريبة جدا من PHP منهم يشبه الآخر: فمثلا لغة للقيام بمواقع و ليس ببرامج

وجدت منذ القديم, مما يجعل منها نموذجا C في الحقيقة لغة للغات أخرى جديدة.
أغلبية لغات البرمجة لديهم في آخر المطاف صفات متشابهة, تتشابه في الأساسيات.

بالتحدث عن الأساسيات : نحن الآن داخل الخليط. رأينا كيفية إنشاء متغيرات, القيام بعمليات بإستعمالهم, سننتقل الآن إلى الإحتمالات.
بدون إحتمالات, برامجنا ستقوم دائما بنفس العمل, مما سيضفي عليها الكثير من الملل

الإحتمال "if... else" booléens, قلب الإحتمالات الإحتمال "switch"

الإحتمال "if... else"

تقوم الإحتمالات بمقارنة المتغيرات. فيمكننا أن نقول"إذا كان يساوي القيمة 50, قم بهذا"...و لكن سيكون من i المتغير الممل أن لا نقوم بمقانة سوى التساوي ! يجب أيضا أن نقارن إذا كان المتغير أصغر من 50 , أصغر من أو يساوي 50, أكبر من, أكبر من أو يساوي ...

حضرت الكل C لا تقلقوا فلغة

سنتبع المخطط, "if... else" لتعلم كيفية إستخدام الإحتمال التالي:

1/ بعض الرموز التي يجب معرفتها قبل البدأ.
2/ المقارن if
3/ المقارن else
4/ المقارن "else if"
5/ عدة إحتمالات في نفس الوقت

هناك على C, في "if... else" قبل البدأ بكتابة إحتمال من النوع 2-3 رموز قاعدية يجب معرفتها. هذه الرموز تعتبر مهمة للقيام بإحتمال ما .

بعض الرموز التي يجب معرفتها قبل البدأ

الرمز	معنـــاه
==	يساوي
>	أكبر قطعا
<	أصغر قطعا
>=	أكبر من أو يساوي
<=	أصغر من أو يساوي
!=	يخالف

**إنتبهوا جيدا, يوجد رمزان "==" لمقارنة التساوي. خطأ شائع يرتكبه المبتدؤون الذي يتجلى في وضع رمز واحد =, الذي ليس له نفس الدلالة في لغة C.سنتحدث عن ذلك في الأسفل.**

المقارن if

فلنهاجم الآن المهم بدون أي إنتظار
سنقوم بإحتمال بسيط, و الذي سيقول للحاسوب :

كود PHP:

```
إذا كان المتغير يساوي قيمة معينة
إذن قم بالآتي
```

و هذا هو اللفظ المستعمل. "if" بالإنجليزية, الكلمة "إذا" تترجم للقيام بإحتمال.
إفتحوا القوسين : داخل داخلهما يجب عليكم كتابة .if أكتبوا الشرط.

بعد ذلك إفتحوا { و أغلقوها بعيدا }. كل ما هو موجود داخل المزدوجتين سيتم تنفيذه فقط إذا تحقق الشرط الموجود داخل القوسين.

و نخلص بعد كل ذلك بهذا الكود:

كود PHP:

```
if(//شرطكم)
{
  // الأوامر المراد تنفيذها إذا تحقق الشرط
}
```

فلنضع مكان التعليق "شرطكم", سنكتب شرطا لكي نقارن متغيرا ما.
الذي يحتوي على عمركم. "age" مثلا, يمكننا مقارنة المتغير
الآن سنقوم بمعرفة هل أنت راشد , أي هل عمرك أكبر من 18 سنة...

كود PHP:

```
if (age >= 18)
{
  printf ("أنت راشد !");
}
```

الرمز => يعني أكبر من أو يساوي, كما رأيناه في الجدول أعلاه.



إذا وجد أمر واحد بين {} فهي تصبح إختيارية فيمكننا كتابة:

كود PHP:

```
if (age >= 18)
  printf ("أنت راشد !");
```

فلنجرب هذا الكود

إذا كنتم تريدون تجربة الأكواد السابقة و ذلك لمعرفة كيفية
و لا ننسى أيضا main وسط الدالة ifيجب وضع ال if, إشتغال
و التي سنعطيها قيمة من إختيارنا age إستدعاء المتغير.
قد تبدوا هذه التجربة سخيفة في نظر البعض لكنها مهمة جدا
فهي ستساعد على ترسيخ مفهوم الإحتمال و كيفية إشتغاله.

إليكم كودا كاملا يمكنكم تجريبة:

كود PHP:

```
#include <stdio.h>
#include <stdlib.h>

int main(int argc, char *argv[])
{
    long age = 20;

    if (age >= 18)
    {
      printf ("Vous etes majeur !\n");
    }

    system("PAUSE");
    return 0;
}
```

يساوي 20 إذن فسيتم القيام بالأمر الموجود age هنا المتغير
ستظهر على "Vous êtes majeur !" داخل {} أي أن الرسالة
الشاشة.
غيروها مثلا للقيمة 10: . age حاولوا الآن تغيير قيمة المتغير
. الشرط سيكون خاطئا و بالتالي الرسالة لن تظهر

إعتمدوا على هذا الكود لتقارنوا الأكواد القادمة...

المقارن "else"

else تعني و إلا...

الآن نعرف كيفية القيام بمقارنة بسيطة, فلنذهب بعض الشيئ بعيدا: إذا لم تنجح المقارنة (الشرط خطأ), يجب أن نقول للحاسوبنا أن يقوم بأمور أخرى instructions.

إذا سنكتب :

كود PHP:

```
إذا كان هذا المتغير== هذه القيمة
قم بهذا
و إلا فقم بهذا...
```

بعد } و سيصبح لدينا else لا يجب علينا سوى إضافة:

كود PHP:

```
if (age >= 18) // إذا كان العمر أكبر من أو يساوي
{
  printf ("أنت راشد !");
}
else // و إلا...
{
  printf ("أنت قاصر !");
}
```

أظن أن هذا الكود مفهوم للجميع...
إذا لم تفهم شيئا لا تترد في طرحه فنحن هنا لمساعدتك...

المقارن "else if"

else if تعني و إلا إذا...

و تتيح لنا لغة سي أيضا جمعهما. if و else رأينا كيف نقوم ب else if... في تعبير واحد و هو
و ذلك للقيام بمقارنة أخرى إذا لم يتحقق الشرط الأول.
else if توجد بين if و else

فنكتب:

كود PHP:

```
إذا كان هذا المتغير== هذه القيمة
قم بهذا
```

و إلا إذا كانت == هذه القيمة
قم بالآتي

و... إلا فقم بهذا

فيصبح لدينا كود على الشكل التالي:
كود PHP:

```
if (age >= 18) // أكبر من أو يساوي 18 age إذا كان
{
  printf ("أنت راشد !");
}
else if ( age > 4 ) // أكبر قطعا من 4 age و إلا إذا كان
{
  printf ("لا زلت صغيرا...");
}
else // و... إلا
{
  printf ("Aga gaa aga gaaa gaaa"); )
}
```

يقوم الحاسوب بالإختبارات بالترتيب:

1. الأول: إذا كان الشرط صحيحا يقوم if في الأول يقوم ب بالأمر الموجود داخل المعقوفتين الأوليتين.

2. و يقوم بإختبار "else if"إذا لم يتحقق الشرط الأول يذهب ل جديد: إذا كان الشرط صحيحا ينجز الأمر الموجود بين المعقوفات.

3. و أخيرا إذا لم يتحقق أي شرط من الشروط السابقة يقوم بأوامر "else"

"else" و "else if" ليسا إجباريين.للقيام بإحتمال لا نحتاج على أقل تقدير if سوى ل
فيمكننا كتابة "else if" لاحظوا أن بإمكاننا إستعمال الكثير من:

كود PHP:

```
if(...)
//الأمر1
else if(...)
//الأمر2
else if(...)
//الأمر3
else if(...)
//الأمر4
else if(...)
```

```
//الأمر5
else(...)
//الأمر6
```

عدة إحتمالات

واحد. مثلا, if قد يكون أيضا القيام بعدة إحتمالات في داخل
. تريدون أن تعرفوا إذا ما كان العمر أكبر من 18 و أصغر من 25

لهذا, يجب إستعمال رموز جديدة:

&&------> "و"
||------> "أو"
!------> "نفي"

&&

إذا كنا نريد أن نقوم بالإحتمال المذكور أعلاه, يجب كتابة:

كود PHP:

```
if (age > 18 && age < 25)
```

العلامتان && يعنيان "و". الإحتمال سيقرأ بالعربية:"إذا كان العمر أكبر من 18 و العمر أصغر من 25".

||

للقيام ب "أو" يجب إستعمال الرمزين ||.
و للقيام بهما فهناك عدة طرق و ذلك حسب نوعية لوحة المفاتيح, إذا ما عليك سوى تجريب إحدى الطرق:





تخيلوا معي برنامجا يقوم بتحديد إذا ما كان الشخص يستطيع

فتح حساب بنكي .
و كما تعرفون لفتح حساب بنكي يجب أن لا تكون صغيرا و لنقل مثلا لا يجب أن يقل عمرك عن 30 سنة أو أن يكون لديك الكثير من المال...
إحتمالنا لمعرفة هل الزبون قادر على فتح حساب بنكي أو لا سيكون كالتالي:

كود PHP:

```
if (age > 30 || money > 100000)
{
  printf("مرحبا  بك في سوبر بنك !");
}
else
{
  printf("أغرب عن وجهي أيها**** !");
}
```

!

هذا الرمز الأخير و الذي هو علامة تعجب يعني في البرمجة "نفي".
يجب وضع هذا الرمز قبل الشرط و ذلك ليكون المعنى: إذا لم هذا الشرط صحيحا.

كود PHP:

```
if (!(age < 18))
```

يمكن ترجمة الكود فوق بالتعبير التالي:
إذا لم يكن الشخص قاصرا.
لو حذفنا الرمز ! لكان معنى الترجمة كالتالي:
إذا كان الشخص قاصرا.

بعض الأخطاء التي يقع فيها الكثيرون

1/ نسيان العلامتين == ووضع واحدة مع أن معناهما مختلف

2/ زيادة النقطة الفاصلة بعد الإحتمال if.

أرجو من الجميع الإنتباه إلى هذه الأخطاء فهي ستسبب لكم الكثير من المشاكل في المستقبل.

Booléens,القلب النابض للإحتمالات.

سندخل الآن في المزيد من التفصيلات حول كيفية عمل الإحتمال if....else.
أضافت الإحتمالات شسئا يسمى ب booléen.

إنه شيئ جد مهم لذلك ففتحوا آذانكم جيدا(نسيت بل إفتحوا أعينكم).

بعض الأمثلة للفهم

إليكم هذا الكود الذيسأطلب منكم تجريبه:
كود PHP:

```
if (1)
{
    printf("C'est vraiصحيح");
}
else
{
    printf("خطأ");
}
```

**لكن ما مهنى هذا؟ لقد وضعت العدد فقط كشرط لهذا الإحتمال؟؟؟**

ستفهمون بعد قليل ...
الآن قوموا بعمل تجربة أخرى لكن بكود مختلف:

كود PHP:

```
if (0)
{
    printf("خطأ");
}
else
{
    printf("صحيح");
}
```

سترون أن النتيجة هي عكس الأولى.
الآن قوموا بعدة تجارب و ذلك بإستبدال 0 بأي قيمة أخرى مثل:
4, 15, 226, -10, -36...

أي كلمة تظهر على الشاشة؟ نجد دائما "صحيح".

إستنتاج: إذا وضعنا الصفر الشرط يكون خاطئا و إذا وضعنا أي عدد آخر يخالف الصفر يكون الشرط صحيحا.

الشرح:

هذا الإحتمال يعيد if في الحقيقة ففي كل مرة تقومون بها ب لنا قيمة 1 إذا كان الشرط صحيحا و 0 إذا كان خاطئا.

مثال:

كود PHP:

```
if (age >= 18)
```

لنفترض age أو من أكبر 23 صحيح فالشرط إذن .23 يساوي بالعدد 1 age >= 18 يساوي 18 فيقوم الحاسوب بتغيير.
و كذلك بالمثل إذا كان الشرط خاطئا يعوضه ب 0.

التجربة بإستعمال متغير

فلنقم بالتجربة كيف نرسخ المفاهيم السابقة: قوموا بإرسال نتيجة الإحتمال إلى متغير, كما لو كانت عملية.

كود PHP:

```
long age = 20;
```

```
int majeur = 0;

majeur = age >= 18;
printf("Majeur = : %ld\n", majeur);
```

كما تلاحظون فإن قيمة المتغير majeur هو 1 .
مثلا هنا ستتغير القيمة age=10 قوموا بنفس العمل لكن بأخذ إلى 0.

المتغير "majeur" يسمى booléen.

نسمي المتغير الذي يأخذ القيم 0 و 1 booléen.
كذلك:

خطأ=0
صحيح=1

لكي نكون دقيقين ف 0 يساوي خطأ و جميع الأعداد الأخرى التي تخالف الصفر صحيحة.
لكن و كما جرت العادة لن نقوم سوى بإستعمال 0 أو 1 .

بالنسبة لنا نحن متعلمي لغة سي فلا يوجد نوع خاص لهذا كنوع خاص .... و "bool" المتغير++لكن تم إضافته في سي هذا ما سنراه إن شاء الله.

للإختصار.bool ب booléen من الآن و صاعدا سأقوم بتغيير

لذلك فإن التوع المستعمل لل bool ك هو أي نوع: char-double-long-int ...
و لا تهتموا بالنوع فقط يجب معرفة كيفية إستخدام bool بطريقة حكيمة.

bool و الإحتمالات

كود PHP:

```
int majeur=1;

if (majeur)
{
  printf("أنت راشد !");
```

```
}
else
{
  printf("أنت قاصر");
}
```

. يحمل القيمة 1 الشرط صحيح majeur بما أن المتغير

إليكم إحتمالا آخر:

كود PHP:

```
if(majeur && man)
```

هذا الإحتمال يترجم كالتالي: إذا كنت راشدا و كنت ذكرا.
إذن **bool** يبين إذا ما كان شيئ صحيح أو خطأ.
ما قمنا به حتى الآن قد يبدو لا فائدة منه لكن بالتعمق في البرمجة نجد أن له إستعمالات كثيرة و مفيدة جداااا...

الإحتمال Switch

الإحتمال if ...else الذي رأيناه من قبل هو الأكثر إستعمالا.
لكن في بعض الأحيان نجد أنفسنا نقوم بالتكرار كمثال الكود أسفله:

كود PHP:

```
if (age == 2)
{
  printf("Salut bebe !");
}
else if (age == 6)
{
  printf("Salut gamin !");
}
else if (age == 12)
{
  printf("Salut jeune !");
}
else if (age == 16)
{
  printf("Salut ado !");
}
else if (age == 18)
{
  printf("Salut adulte !");
}
else if (age == 68)
```

```
{
  printf("Salut papy !");
}
else
{
  printf("Je n'ai aucune phrase de prete pour ton age  ");
}
```

القيام ب switch

لتفادي القيام بعدة إحتمالات على متغير واحد, تم إبتكار طريقة إخرى للقيام بالإحتمالات.
و هذا مثال عليه switch هذه الطريقة تسمى:

كود PHP:

```
switch (age)
{
case 2:
  printf("Salut bebe !");
  break;
case 6:
  printf("Salut gamin !");
  break;
case 12:
  printf("Salut jeune !");
  break;
case 16:
  printf("Salut ado !");
  break;
case 18:
  printf("Salut adulte !");
  break;
case 68:
  printf("Salut papy !");
  break;
default:
  printf("Je n'ai aucune phrase de prete pour ton age  ");
  break;
}
```

نكتب:

switch(المتغير1)
و ذلك للدلاة على أنني سأقوم بعمل الإحتمالات على المتغير1.
نفتح المعقوفات و نغلقها تحت.

و داخل المعقوفات نتعامل مع الحالات الممكنة: case 2, case 4, case 5, case 45...

**بعد كل حالة و هو إجباري إذا لم ;break يجب وضع الأمر**

تقوموا بذلك فالحاسوب سيقرأ جميع الحالات و يقوم بجميع الأوامر...
لذلك حذاري من نسيان هذا الأمر.

إذا لم يحقق else الذي يعمل عمل default و لا ننسى الأمر المتغير1 الحالات المذكورة يقوم الحاسوب بالقيام بأوامر default.

القيام بقائمة للأطعمة :

يقوم السويتش عادة بالقيام بالقائمات في الكونصول.
إذا إلى العمل.

المطلوب منكم:

أن تقوموا بعمل قائمة للأطعمة التي يقدمها مطعمكم و أن تعرضها على المستعمل فيختار منها ما يريد ثم تكتب لقد إخترت ما يلي...مثلا.

ماذا تنتظرون هيا للعمل

الحل:

كود PHP:

```
#include <stdio.h>
#include <stdlib.h>

int main(int argc, char *argv[])
{
  long choixMenu;

  printf("=== Menu ===\n\n");
  printf("1. Royal Cheese\n");
  printf("2. Mc Deluxe\n");
  printf("3. Mc Bacon\n");
  printf("4. Big Mac\n");
  printf("\nVotre choix ? ");
  scanf("%ld", &choixMenu);

  printf("\n");

  switch (choixMenu)
  {
    case 1:
       printf("Vous avez choisi le Royal Cheese. Bon choix !");
       break;
```

```
    case 2:
       printf("Vous avez choisi le Mc Deluxe. Berk, trop de sauc
e...");
       break;
    case 3:
       printf("Vous avez choisi le Mc Bacon. Bon, ca passe encor
e ca ;o)");
       break;
    case 4:
       printf("Vous avez choisi le Big Mac. Vous devez avoir tre
s faim !");
       break;
    default:
       printf("Vous n'avez pas rentre un nombre correct. Vous ne
mangerez rien du tout !");
       break;
  }

  printf("\n\n");

  system("PAUSE");
}
```

و هاقد إنتهينا...
في آخر السويتش **default** أظن أنكم لم تنسوا
في الحقيقة عندما تبرمجون يجب عليكم التفكير في جميع الإحتمالات.
ستجدون دائما من يضغط على الرقم 10 أو سيكتب مرحبا مع أنكم لم تنتظروا ذلك.

لذلك لا تضعوا ثقتكم في المستخدم يستطيع دائما إدخال أي شيئ لذلك يجب أخذ كل شيئ بعين الإعتبار.

بالنسبة للتمارين فسأضعها في درس خاص بها بما أنكم أصبحتم تتقدمون...

سامحوني على تأخر الدرسو ذلك لظروف شخصية
فسامحوني لقد وجدت اليوم متسعا من الوقت فكتبت ما تبقى من الدرس...

إنتهى
و بالتوفيق للجميع

أشهد أن لا إلـه إلا **الـلـه** و أن **محمدا** رسول **الـلـه**